DYNAMIC LOW-RANK MATRIX RECOVERY: THEORY AND APPLICATIONS

A Dissertation
Presented to
The Academic Faculty

By

Liangbei Xu

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

May 2020

DYNAMIC LOW-RANK MATRIX RECOVERY: THEORY AND APPLICATIONS

Approved by:

Dr. Mark A. Davenport, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Christopher J. Rozell
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Justin K. Romberg
School of School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Yao Xie
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Mary Wootters
School of Computer Science
Stanford University

Date Approved: December 9, 2019

Information is the resolution of uncertainty.

Claude Shannon

To my family.

ACKNOWLEDGEMENTS

First, I would like to express my deepest gratitude to my advisor Prof. Mark A. Davenport for his tremendous support, valuable advice, and continuous encouragement along my journey to complete the Ph.D. program. He taught me how to do research independently, including how to define research problems, conduct literature surveys, and propose impactful solutions to problems. In addition, he encouraged me to think critically, gave me freedom to work on topics in which I was interested, and contributed valuable feedback and suggestions. Without his careful guidance and support, I would not have completed this dissertation. I also would like to extend my sincere thanks to my co-advisor Prof. Mary Wootters in the theoretical computer science group at Stanford University. With her valuable advice and help, I really had a great time while visiting Stanford.

I also would like to joyously acknowledge my fellow scholars and alumni: Andy, Michael, Santhosh, Matt, Andrew, Nauman, Namrata, Rakshith, Hongteng, Kyle, Ning, Sohail, Darryl, Tomer, Sihan, John, Adam, Marrisa, Pavel, Greg, and Nick, to name just a few. I have enjoyed all of our interactions, accompanying discussion, and collaboration.

I would like to thank Prof. Romberg, Prof. Rozell, Prof. Wootters, and Prof. Yao for serving on my PhD dissertation committee. Their beneficial comments and invaluable advice helped bring this dissertation to completion. In addition, I would like to thank all the faculty in the Center of Signal and Information Processing at Georgia Tech, especially Prof. Romberg, Prof. Rozell, and Prof. Zhou for their generous advice and help with my research.

I would like to thank my family for their love and support through the Ph.D. journey. Finally, I wish to acknowledge all who have been part of my life during the last five years. Thanks to you, the past five years' stay at Georgia Tech will be the most memorable experience of my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The purpose of this work is to provide both theoretical understanding of and practical algorithms for dynamic low-rank matrix recovery. Although the benefits of exploiting dynamics in low-rank matrix recovery have been observed in many applications, the theoretical understanding of and justification for these methods is limited. This dissertation concerns two widely-used dynamics models in the context of low-rank matrix recovery: random walk dynamics and measurement induced dynamics.

The first part of this dissertation studies the theoretical properties, including recovery guarantees and algorithmic convergence, of dynamic low-rank matrix recovery under a random walk dynamics model. In the proposed locally weighted matrix smoothing (LOWEMS) framework, we provide answers to the following questions: (1) What kind of reduction in sample complexity is possible by exploiting dynamic structure in the underlying matrix? (2) How do the recovery error guarantees compare to the corresponding guarantees for the static baseline cases? We also provide numerical simulations to validate our analysis and real-world experiments to show our methods' empirical effectiveness. Furthermore, we discuss two extensions of LOWEMS: one-bit LOWEMS for binary measurements and S-LOWEMS for quickly and simultaneously recovering a series of low-rank matrices.

Though the random walk dynamics model is effective in practice, it might not be the best way to describe the dynamics of some low-rank matrix recovery problems, such as student learning process. In the second part of this dissertation, we study dynamic low-rank matrix recovery under a measurement induced dynamics model. We first investigate a practical application in the context of knowledge tracing. We present the *DynEmb* framework to track student knowledge in an intelligent tutoring system (ITS) by using techniques from static low-rank matrix recovery and recurrent neural networks (RNNs). We then describe a simple low-rank matrix recovery model drawn from the DynEmb framework and provide recovery guarantees for it. This theoretical analysis not only helps to fill the gap between classical

matrix sensing and matrix completion theory but also provides some initial theoretical analysis for the problem of dynamic low-rank matrix recovery with measurement induced dynamics.

CHAPTER 1
INTRODUCTION AND BACKGROUND

1.1 Motivation

Low-rank matrix models have proved to be useful in many applications, including:

- Linear system identification: low-rank (Hankel) matrices correspond to low-order linear time-invariant systems [1, 2]
- Signal processing: source separation [3] and blind deconvolution [4, 5, 6]
- Recommendation systems: Netflix challenge [7] and Yahoo music contest [8]
- Randomized linear algebra: sketching as a tool for numerical linear algebra [9]

In these and many other applications, the data matrix is often extremely large, so that it can be impossible even to observe all of the entries of the matrix, let alone perform traditional processing steps in a computationally realistic manner. However, in many cases the physical system underlying the data matrix is relatively simple, resulting in a great deal of structure in the data matrix. For example, consider the student learning process. We have a set of n students and a collection of m questions, and let $X \in \mathbb{R}^{m \times n}$ denote the response matrix. We assume each question is only related to a small number of abstract concepts. A common assumption is that X can be factored as $X = UV^T$, where $U \in \mathbb{R}^{m \times r}$ is a matrix relating the m questions to the r concepts, and $V \in \mathbb{R}^{n \times r}$ represents the students' knowledge of the underlying concepts. Though it may be impossible to have all students answer each question, harnessing such structural properties enable us to reconstruct the matrix.

In applications, it has also been widely noted that by incorporating temporal information and allowing for the possibility of time-varying signals, significant improvements over static

models are possible in practice. For example, in recommendation systems, users' preferences for various items may change (sometimes quite dramatically) over time. Modeling such drift has been proposed for both music and movies as a way to achieve higher accuracy [10, 11]. Another example in signal processing is dynamic non-negative matrix factorization for the blind signal separation problem [12].

Although there has been much progress recently in our theoretical understanding of low-rank matrix recovery from a few measurements in the static case (see [13] and [14] for an overview), there is limited theoretical justification for introducing more complex dynamic models, despite their superior empirical performance. In this dissertation we aim to address this gap by studying the problem of recovering a dynamically evolving low-rank matrix from incomplete observations. We first summarize some prototypical dynamic low-rank matrix models that arise in real world problems, and then we turn to address some of the following fundamental questions:

1. How many measurements are required to reconstruct a dynamic low-rank matrix?

2. Is reconstruction stable under a realistic noise model?

3. Is there an efficient algorithm to perform the reconstruction?

4. What is the performance in a real world implementation compared to static baselines?

We answer these questions using two prototypical dynamics models in the context of low-rank matrix recovery: random walk dynamics and measurement induced dynamics.

1.2 Contributions

The first contribution of this work is to extend recovery guarantees for low-rank matrix recovery to the setting of the random walk dynamics. We present this work in Chapter 2. The analysis consists of two parts: (1) recovery guarantees for matrix sensing and matrix completion; and (2) convergence guarantees for gradient descent algorithms. The second

contribution, presented in Chapter 3, is to propose two extensions of LOWEMS: one-bit LOWEMS for binary measurements and S-LOWEMS for fast recovery of a series of low-rank matrices.

In Chapter 4, we consider the setting in which the dynamics of the underlying low-rank matrix are caused in part by the measurement process. Our third contribution is first to develop the *DynEmb* framework to harness techniques from low-rank matrix recovery and recurrent neural networks to track effectively the underlying dynamic low-rank matrix and second to demonstrate the effectiveness of this framework in a knowledge tracing application. In analyzing the theoretical properties of dynamic low-rank matrix recovery with measurement induced dynamics, we derive a novel simplified sampling scheme and then prove a recovery guarantee for it in Chapter 5. This analysis not only helps to fill the gap between classical matrix sensing and matrix completion theory but also provides initial insights into the theoretical understanding of dynamic low-rank matrix recovery under a measurement induced dynamics model.

1.3 Background

1.3.1 Notation

Before proceeding, we briefly state some of the notation that we will use throughout. For a vector $x \in \mathbb{R}^n$, we let $\|x\|_p$ denote the standard ℓ_p norm. Given a matrix $X \in \mathbb{R}^{n_1 \times n_2}$, we use $X_{i:}$ to denote the i^{th} row of X and $X_{:j}$ to denote the j^{th} column of X. We let $\|X\|_F$ denote the the Frobenius norm, $\|X\|_2$ the operator norm, $\|X\|_*$ the nuclear norm, and $\|X\|_\infty = \max_{i,j} |X_{ij}|$ the elementwise ℓ_∞ norm. Given a pair of matrices $X, Y \in \mathbb{R}^{n_1 \times n_2}$, we let $\langle X, Y \rangle = \sum_{i,j} X_{ij} Y_{ij} = \text{Tr}\left(Y^T X\right)$ denote the standard inner product. Finally, we let $n_{\max}$ and $n_{\min}$ denote $\max\{n_1, n_2\}$ and $\min\{n_1, n_2\}$ respectively. We use $\mathcal{N}(0, 1)$ to denote the standard Gaussian distribution and $\text{Ber}(p)$ the Bernoulli distribution with success probability p. We use $f = \Theta(g)$ to denote that f is bounded both above and below by g asymptotically, and by $f \gtrsim g$ we mean that f is greater than g up to a constant.

1.3.2 Compressive sensing

Many signals, including images, videos, acoustic signals, user-item response data, and medical data, are compressible in that they can be well-approximated by a combination of a few atoms from an appropriate dictionary; this can be exploited to yield advantages in both computation and storage. Though this idea has attracted significant attention recently, people were already aware of compressible signals as early as when Joseph Fourier initiated the study of Fourier series and their applications. In 1795, Prony proposed an algorithm for the estimation of the parameters associated with a small number of complex exponentials sampled in the presence of noise. This work contains the core idea of compressive sensing: recovering a signal from a limited number of measurements. More recently, Candes, Romberg, Tao, [15, 16, 17, 18, 19] and Donoho [20, 21] showed that a compressible (or sparse) signal can, with high probability, be reconstructed exactly or approximately by a small set of linear, non-adaptive measurements using polynomial time algorithms. These results suggest that we can recover a sparse signal using far fewer measurements than are required by the classical Nyquist-Shannon framework, hence the name *compressive sensing*.

Mathematical fundamentals of compressive sensing

We say that a vector $x \in \mathbb{R}^n$ is s-sparse if at most s of its entries are nonzero, i.e., if $\|x\|_0 \leq s$. We want to reconstruct x from the following linear measurements:

$$y = Ax,$$

where $y \in \mathbb{R}^m$ is the vector of measurements and $A \in \mathbb{R}^{m \times n}$ is the measurement matrix. The core idea of compressive sensing is to reconstruct x by the following ℓ_1-minimization program:

$$\hat{x}_{\ell_1} = \arg\min_x \|x\|_1 \quad \text{subject to } Ax = y \tag{1.1}$$

instead of the ℓ_0-minimization program

$$\hat{x}_{\ell_0} = \arg\min_x \|x\|_0 \quad \text{subject to } Ax = y. \tag{1.2}$$

The ℓ_0-minimization program is non-convex and NP-hard (proved in [22]). However, ℓ_1 minimization is convex and can be computed in polynomial time. Although empirical observations that ℓ_1 minimization promotes sparse solutions were reported in various areas since the 1970s [23], strong theoretical results were just recently established in 2004 by Candes, Donoho, Romberg and Tao [15, 16, 17, 18, 19, 20, 21]. This work can be summarized in the following theorem:

Theorem 1.3.1. *For any s-sparse vector $x_0 \in \mathbb{R}^n$, if $A \in \mathbb{R}^{m\times n}$ contains i.i.d. standard Gaussian entries and $m \gtrsim n\log(n/s)$, then x_0 is the unique solution to program* (1.1) *with probability at least* $1 - \Theta(e^{-m})$.

The core mathematical tool used to establish Theorem 1.3.1 is *concentration of measure* associated with the independent Gaussian random measurements; this can be further generalized to sub-Gaussian matrix entries and other, more structured random measurements. The proof procedure is to establish the *restricted isometric property* (RIP) for the Gaussian random measurements with high probability and show that the RIP leads to uniqueness of program (1.1). Note that Theorem 1.3.1 establishes solution uniqueness for *all* s-sparse signals; this is a *uniform* recovery guarantee. When the solution uniqueness is established for a *fixed* s-sparse signal, we call this a *non-uniform* recovery guarantee.

There are other mathematical approaches for establishing uniform or non-uniform recovery guarantees for various structured signals from other structured random measurements. For example, [24] introduces the *incoherence* between sensing modality and signal basis to handle measurements selected from an orthonormal basis. Moreover, [25] uses Gordon's Escape Through a Mesh Theorem [26] and convex geometric analysis to establish a non-uniform exact recovery guarantee for a set of low-dimensional models. Candes

and Plan [27] establish a non-uniform recovery guarantee for sparse vectors under general random measurements.

1.3.3 Low-rank matrix recovery

Although an $n_1 \times n_2$ matrix can be considered as a vector in $\mathbb{R}^{n_1 \times n_2}$, low-dimensional properties like sparsity are not enough to capture the special structure of low-rank matrices. Suppose that $X \in \mathbb{R}^{n_1 \times n_2}$ is a rank-r matrix with r much smaller than n_1 and n_2. We observe X through a linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$,

$$y = \mathcal{A}(X), \quad y \in \mathbb{R}^m.$$

Many data matrices in the real world are (approximately) low-rank. For example, the covariance matrix of a linear dynamical system is approximately of rank r, where r is the dimension of the state variable. In penalization learning, the student-question response matrix is approximately low-rank, if we assume that students' knowledge state are decided by a few hidden factors.

In practice, one often cannot observe all the entries of X explicitly. There are several forms of the measurement $\mathcal{A}$ that have attracted attention recently. When $\mathcal{A}$ is a set of weighted linear combinations of the entries of X, this problem is often called the *matrix sensing* problem. The case in which $\mathcal{A}$ samples a subset of entries of X is known as the *matrix completion* problem. One might consider matrix completion as a special case of matrix sensing; however, the methods of theoretical analysis are quite different.

Mathematical fundamentals of low-rank matrix recovery

Consider the following linear measurement operator:

$$[\mathcal{A}(X)]_i = \langle A_i, X \rangle, i = 1, \ldots, m.$$

In a similar manner to the vector case, one can reconstruct X by the following convex *nuc-min* program:

$$\hat{X}_{nuc} = \arg\min_{X} \|X\|_* \quad \text{subject to } \mathcal{A}(X) = y \tag{1.3}$$

instead of the *rank-min* program:

$$\hat{X}_{rank} = \arg\min_{X} \text{rank}(X) \quad \text{subject to } \mathcal{A}(X) = y. \tag{1.4}$$

The rank-min program is non-convex and NP-hard (proved in [28]). However, the nuc-min program (first proposed in [29]) is convex and can be further reformulated as a semidefinite program (SDP), which can be solved by many off-the-shelf polynomial-time algorithms, such as the interior point method.

Applying mathematical tools similar to those used in compressive sensing, Recht, Fazel and Parrilo [30] established a seminal recovery guarantee for the nuc-min program from random measurements, which is further strengthened in [31]. Their results are summarized in the following theorem:

Theorem 1.3.2. *If $A_i \in \mathbb{R}^{n_1 \times n_2}$, for $i = 1, \ldots, m$, contains i.i.d. standard Gaussian entries, and $m \gtrsim r(n_1 + n_2)$, then, for any rank-r matrix $X_0 \in \mathbb{R}^{n_1 \times n_2}$, X_0 is the unique solution to program* (1.3) *with probability at least $1 - \Theta(e^{-m})$.*

The proof procedure is to establish a high-probability matrix RIP for Gaussian random measurements, and then to show that the matrix RIP leads to solution uniqueness of program (1.3).

Another related problem of particular interest is matrix completion, which assumes that each A_i samples one entry of X uniformly at random, either with or without replacement. For matrix completion, the matrix RIP does not hold. However, one can still show a non-uniform recovery guarantee:

Theorem 1.3.3 ([32]). *Let $X \in \mathbb{R}^{n_1 \times n_2}$ be a fixed matrix of rank r obeying the strong incoherence property with parameter μ. Suppose we observe m entries of X with locations sampled uniformly at random (with or without replacement). Then, if $m \gtrsim \mu^2 nr \log^6 n$, X is the unique solution to* (1.3) *with probability at least $1 - \Theta(n^{-3})$.*

The strong incoherence μ measures the similarity between the sensing modality and the subspace spanned by X, which is a similar concept to that in the vector case (see [24]), and one can refer to [32] for an explicit definition. The sample complexity required is further reduced to $\mu^2 nr \log^2 n$ in [33].

There are other approaches for establishing uniform and non-uniform guarantees for various low-rank matrix recovery problems. For example, Keshavan et al. [34] propose the OPTSPACE algorithm for matrix completion and establishes its convergence, along with a recovery guarantee. Gross [35] and later Recht [36] use the "golfing scheme" to construct a dual certificate for the nuc-min program (1.3) in matrix completion and hence show its recovery guarantee.

There are many other structured measurement models for low-rank matrix recovery, such as blind deconvolution [5] and phase retrieval [37]. See [13] for an overview.

1.3.4 Dynamic low-rank matrix recovery

Nearly all of this existing work assumes that the underlying low-rank matrix X remains fixed throughout the measurement process. However, in many practical applications, this is a major limitation. For example, in a recommendation system, users' preferences for various items may change (sometimes quite dramatically) over time [10, 11]. In collaborative filtering, approaches for dealing with the dynamics of users' preferences over time can generally be categorized into two types [38]. The first type basically contains *time-aware* algorithms (see [39] for an overview of time-aware models), which use the time information as static features or context to better capture people's preference at different times. For example, users' preferences for summer/winter clothes vary with seasons and users' preferences for

music vary with weekday/weekend. On the other hand, users' preferences for various styles of clothes is changing over time due to fashion trends, users' aging, and other non-stationary factors. The second type of approach is called the *time-changing* approach, which is also named *concept drift* [40] in cognitive science. Throughout this dissertation, we focus on modeling time-changing dynamics using low-rank matrices.

In [41], the authors propose an online algorithm for a dynamic exponential matrix factorization model and show its superior performance on e-mail data over static baselines. The authors assume both factor matrices follow random walk dynamics. Similarly, [42] provides a Kalman-filtering-style algorithm by assuming that only users' preferences follow random walk dynamics. In [43], the authors propose an exponentially-decaying smoothing technique to exploit the dynamics in a non-negative matrix factorization (NMF) model and demonstrate its superior performance over baselines for the detection of Alzheimer's disease. A similar weighting schemes is also found in [44]. Sun et al. [45] propose using a transition matrix to model the dynamics of users' preferences, and they develop an expectation-maximization (EM) algorithm to recover both factor matrices and transition matrices. A similar model is also found in [12].

Previous results on dynamic low-rank matrix recovery are almost entirely empirical. There is limited theoretical analysis and justification for these models and algorithms. In this dissertation we aim to summarize some prototypical dynamics models and provide both theoretical analysis and real-world experiments to bridge the gap between practice and theory.

CHAPTER 2

LOW-RANK MATRIX SMOOTHING FOR RANDOM WALK DYNAMICS

Low-rank matrix factorizations arise in a wide variety of applications, including recommendation systems, topic models, and source separation. In these and many other applications, it has been widely noted that by incorporating temporal information and allowing for the possibility of time-varying matrices, significant improvements over static models are possible in practice. However, despite the empirical success of these dynamic models, there is currently limited theoretical understanding of them. In this chapter we aim to address this gap by studying the problem of recovering from incomplete observations a dynamically evolving low-rank matrix under a random walk dynamic model. First, we propose the locally weighted matrix smoothing (LOWEMS) framework for dynamic matrix recovery. We then establish error bounds for LOWEMS in both the *matrix sensing* and *matrix completion* observation models. Our results quantify the potential benefits of exploiting dynamics constraints both in terms of recovery accuracy and sample complexity. To illustrate these benefits we provide both synthetic and real-world experimental results.

2.1 Introduction

Suppose that $X \in \mathbb{R}^{n_1 \times n_2}$ is a rank-r matrix with r much smaller than n_1 and n_2. We observe X through a linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$,

$$y = \mathcal{A}(X), \quad y \in \mathbb{R}^m.$$

In recent years there has been a significant amount of progress in our understanding of how to recover X from observations of this form even when the number of observations m is

Material in this section is joint work with Mark Davenport, leading to publication [46].

much less than the number of entries in X. (See [13] for an overview of this literature.) When $\mathcal{A}$ is a set of weighted linear combinations of the entries of X, this problem is often referred to as the *matrix sensing* problem. In the special case where $\mathcal{A}$ samples a subset of entries of X, it is known as the *matrix completion* problem. There are a number of ways to establish recovery guarantees in these settings. Perhaps the most popular approach for theoretical analysis in recent years has focused on the use of nuclear norm minimization as a convex surrogate for the (nonconvex) rank constraint [47, 48, 31, 49, 32, 50, 51, 52, 30, 53]. An alternative, however is to aim to directly solve the problem under an exact low-rank constraint. This leads a non-convex optimization problem, but has several computational advantages over most approaches to minimizing the nuclear norm and is widely used in large-scale applications (such as recommendation systems) [7]. In general, popular algorithms for solving the rank-constrained models – e.g., alternating minimization and alternating gradient descent – do not have as strong of convergence or recovery error guarantees due to the non-convexity of the rank constraint. However, there has been significant progress on this front in recent years [54, 55, 56, 57, 58, 59, 60], with many of these algorithms now having guarantees comparable to those for nuclear norm minimization.

Nearly all of this existing work assumes that the underlying low-rank matrix X remains fixed throughout the measurement process. In many practical applications, this is a tremendous limitation. For example, users' preferences for various items may change (sometimes quite dramatically) over time. Modeling such drift of user's preference has been proposed in the context of both music and movies as a way to achieve higher accuracy in recommendation systems [10, 11]. Another example in signal processing is dynamic non-negative matrix factorization for the blind signal separation problem [12]. In these and many other applications, explicitly modelling the dynamic structure in the data has led to superior empirical performance. However, our theoretical understanding of dynamic low-rank matrix recovery is still very limited.

We provide the first theoretical results on the dynamic low-rank matrix recovery problem.

We determine the sense in which dynamic constraints can help to recover the underlying time-varying low-rank matrix in a particular dynamic model and quantify this impact through recovery error bounds. To describe our approach, we consider a simple example where we have two rank-r matrices X^1 and X^2. Suppose that we have a set of observations for each of X^1 and X^2, given by

$$y^i = \mathcal{A}^i\left(X^i\right), \quad i = 1, 2.$$

The naïve approach is to use y^1 to recover X^1 and y^2 to recover X^2 separately. In this case the number of observations required to guarantee successful recovery is roughly $m^i \geq C^i r \max(n_1, n_2)$ for $i = 1, 2$ respectively, where C^1, C^2 are fixed positive constants (see [31]). However, if we know that X^2 is close to X^1 in some sense (for example, if X^2 is a small perturbation of X^1), then the above approach is suboptimal both in terms of recovery accuracy and sample complexity, since in this setting y^1 actually contains information about X^2 (and similarly, y^2 contains information about X^1). There are a variety of possible approaches to incorporating this additional information. The approach we will take is inspired by the LOWESS (locally weighted scatterplot smoothing) approach from non-parametric regression. In the case of this simple example, if we look just at the problem of estimating X^2, our approach reduces to solving a problem of the form

$$\min_{X^2} \|\mathcal{A}^2(X^2) - y^2\|_2^2 + \lambda \|\mathcal{A}^1(X^2) - y^1\|_2^2 \qquad \text{s.t.} \quad \text{rank}\left(X^2\right) \leq r,$$

where λ is a parameter that determines how strictly we are enforcing the dynamic constraint (if X^1 is very close to X^2 we can set λ to be larger, but if X^1 is far from X^2 we will set it to be comparatively small). This approach generalizes naturally to the *locally weighted matrix smoothing* (LOWEMS) program described in Section 2.2. Note that it has a (simple) convex objective function, but a non-convex rank constraint. Our analysis in Section 2.3 shows that the proposed program outperforms the above naïve recovery strategy both in terms of recovery accuracy and sample complexity.

We should emphasize that the proposed LOWEMS program is non-convex due to the exact low-rank constraint. Inspired by previous work on matrix factorization, we propose using an efficient alternating minimization algorithm (described in more detail in Section 2.4). We explicitly enforce the low-rank constraint by optimizing over a rank-r factorization and alternately minimize with respect to one of the factors while holding the other one fixed. This approach is popular in practice since it is typically less computationally complex than nuclear norm minimization based algorithms. In addition, thanks to recent work on global convergence guarantees for alternating minimization for low-rank matrix recovery [55, 57, 60], one can reasonably expect similar convergence guarantees to hold for alternating minimization in the context of LOWEMS, although we leave the pursuit of such guarantees for future work.

To empirically verify our analysis, we perform both synthetic and real world experiments, described in Section 2.5. The synthetic experimental results demonstrate that LOWEMS outperforms the naïve approach in practice both in terms of recovery accuracy and sample complexity. We also demonstrate the effectiveness of LOWEMS in the context of recommendation systems.

2.2 Problem formulation

The underlying assumption throughout this Chapter is that our low-rank matrix is changing over time during the measurement process. For simplicity we will model this through the following discrete dynamic process: at time t, we have a low-rank matrix $X^t \in \mathbb{R}^{n_1 \times n_2}$ with rank r, which we assume is related to the matrix at previous time-steps via

$$X^t = f(X^1, \ldots, X^{t-1}) + \epsilon^t,$$

where ϵ^t represents noise. Then we observe each X^t through a linear operator $\mathcal{A}^t : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{m^t}$,

$$y^t = \mathcal{A}^t(X^t) + z^t, \quad y^t, z^t \in \mathbb{R}^{m^t}, \tag{2.1}$$

where z^t is measurement noise. In our problem we will suppose that we observe up to d time steps, and our goal is to recover $\{X^t\}_{t=1}^d$ jointly from $\{y^t\}_{t=1}^d$.

The above model is sufficiently flexible to incorporate a wide variety of dynamics, but we will make several simplifications. First, we note that we can impose the low-rank constraint explicitly by factorizing X^t as $X^t = U^t \left(V^t\right)^T, U^t \in \mathbb{R}^{n_1 \times r}, V^t \in \mathbb{R}^{n_2 \times r}$. In general both U^t and V^t may be changing over time. However, in some applications, it is reasonable to assume that only one set of factors is changing. For example, in a recommendation system where our matrix represent user preferences, if the rows correspond to items and the columns correspond to users, then U^t contains the latent properties of the items and V^t models the latent preferences of the users. In this context it is reasonable to assume that only V^t changes over time [10, 11], and that there is a fixed matrix U (which we may assume to be orthonormal) such that we can write $X^t = UV^t$ for all t. Similar arguments can be made in a variety of other applications, including personalized learning systems, blind signal separation, and more.

Second, we assume a Markov property on f, so that X^t (or equivalently, V^t) only depends on the previous X^{t-1} (or V^{t-1}). Furthermore, although other dynamic models could be accommodated, for the sake of simplicity in our analysis we consider the simple model on V^t where

$$V^t = V^{t-1} + \epsilon^t, \quad t = 2, \ldots, d. \tag{2.2}$$

We will also assume that both ϵ^t and the measurement noise z^t are i.i.d. zero-mean Gaussian noise.

To simplify our discussion, we will assume that our goal is to recover the matrix at the most recent time-step, i.e., we wish to estimate X^d from $\{y^t\}_{t=1}^d$. Our general approach

can be stated as follows. The LOWEMS estimator is given by the following optimization program:

$$\hat{X}^d = \underset{X \in \mathbb{C}(r)}{\arg\min}\, \mathcal{L}(X) = \underset{X \in \mathbb{C}(r)}{\arg\min}\, \frac{1}{2} \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(X) - y^t \right\|_2^2, \tag{2.3}$$

where $\mathbb{C}(r) = \{X \in \mathbb{R}^{n_1 \times n_2} : \text{rank}(X) \leq r\}$, and $\{w_t\}_{t=1}^d$ are non-negative weights. We further assume $\sum_{t=1}^d w_t = 1$ to avoid ambiguity. In the following section we provide bounds on the performance of the LOWEMS estimator for two common choices of operators $\mathcal{A}^t$.

2.3 Recovery error bounds

Given the estimator $\hat{X}^d$ from (2.3), we define the recovery error to be $\Delta^d := \hat{X}^d - X^d$. Our goal in this section will be to provide bounds on $\|\hat{X}^d - X^d\|_F$ under two common observation models. Our analysis builds on the following (deterministic) inequality.

Proposition 2.3.1. *Both the estimator $\hat{X}^d$ by* (2.3) *and* (2.9) *satisfies*

$$\sum_{t=1}^{d} w_t \left\| \mathcal{A}^t\left(\Delta^d\right) \right\|_2^2 \leq 2\sqrt{2r} \left\| \sum_{t=1}^{d} w_t \mathcal{A}^{t*}\left(h^t - z^t\right) \right\|_2 \left\| \Delta^d \right\|_F, \tag{2.4}$$

where $h^t = \mathcal{A}^t\left(X^d - X^t\right)$ and $\mathcal{A}^{t*}$ is the adjoint operator of $\mathcal{A}^t$.

This is a deterministic result that holds for any set of $\{\mathcal{A}^t\}$. The remaining work is to lower bound the LHS of (2.4), and upper bound the RHS of (2.4) for concrete choices of $\{\mathcal{A}^t\}$. In the following sections we derive such bounds in the settings of both Gaussian matrix sensing and matrix completion. For simplicity and without loss of generality, we will assume $m^1 = \ldots = m^d =: m_0$, so that the total number of observations is simply $m = dm_0$.

2.3.1 Matrix sensing setting

For the matrix sensing problem, we will consider the case where all operators $\mathcal{A}^t$ correspond to Gaussian measurement ensembles, defined as follows.

Definition 2.3.2. [31] A linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$ is a Gaussian measurement ensemble if we can express each entry of $\mathcal{A}(X)$ as $[\mathcal{A}(X)]_i = \langle A_i, X \rangle$ for a matrix A_i whose entries are i.i.d. according to $\mathcal{N}(0, 1/m)$, and where the matrices $A_1, \ldots, A_m$ are independent from each other.

Also, we define the matrix restricted isometry property (RIP) for a linear map $\mathcal{A}$.

Definition 2.3.3. [31] For each integer $r = 1, \ldots, n_{\min}$, the isometry constant δ_r of $\mathcal{A}$ is the smallest quantity such that

$$(1 - \delta_r) \|X\|_F^2 \leq \|\mathcal{A}(X)\|_2^2 \leq (1 + \delta_r) \|X\|_F^2$$

holds for all matrices X of rank at most r.

An important result (that we use in the proof of Theorem 2.3.4) is that Gaussian measurement ensembles satisfy the matrix RIP with high probability provided $m \geq Cr n_{\max}$. See, for example, [31] for details.

To obtain an error bound in the matrix sensing case we lower bound the LHS of (2.4) using the matrix RIP and upper bound the stochastic error (the RHS of (2.4)) using a covering argument. The following is our main result in the context of matrix setting.

Theorem 2.3.4. *Suppose that we are given measurements as in* (2.1) *where all $\mathcal{A}^t$'s are Gaussian measurement ensembles. Assume that X^t evolves according to* (2.2) *and has rank r. Further assume that the measurement noise z^t is i.i.d. $\mathcal{N}(0, \sigma_1^2)$ for $1 \leq t \leq d$ and that the perturbation noise ϵ^t is i.i.d. $\mathcal{N}(0, \sigma_2^2)$ for $2 \leq t \leq d$. If*

$$m_0 \geq D_1 \max \left\{ n_{\max} r \sum_{t=1}^{d} w_t^2, n_{\max} \right\}, \tag{2.5}$$

where D_1 is a fixed positive constant, then the estimator $\hat{X}^d$ from (2.3) *satisfies*

$$\left\|\Delta^d\right\|_F^2 \leq C_0 \left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right) n_{\max} r \tag{2.6}$$

with probability at least $P_1 = 1 - dC_1 \exp(-c_1 n_2)$*, where* C_0, C_1, c_1 *are positive constants.*

If we choose the weights as $w_d = 1$ and $w_t = 0$ for $1 \leq t \leq d-1$, the bound in Theorem 2.3.4 reduces to a bound matching classical (static) matrix recovery results (see, for example, [31] Theorem 2.4). Also note that in this case Theorem 2.3.4 implies exact recovery when the sample complexity is $O(rn/d)$. In order to help interpret this result for other choices of the weights, we note that for a given set of parameters, we can determine the optimal weights that will minimize this bound. Towards this end, we define $\kappa := \sigma_2^2/\sigma_1^2$ and set $p_t = (d-t), 1 \leq t \leq d$. Then one can calculate the optimal weights by solving the following quadratic program:

$$\{w_t^*\}_{t=1}^d = \underset{\sum_t w_t = 1;\ w_t \geq 0}{\arg\min} \sum_{t=1}^{d} w_t^2 + \sum_{t=1}^{d-1} p_t \kappa w_t^2. \tag{2.7}$$

Using the method of Lagrange multipliers one can show that (2.7) has the analytical solution:

$$w_j^* = \frac{1}{\sum_{i=1}^d \frac{1}{1+p_i\kappa}} \frac{1}{1+p_j\kappa}, \quad 1 \leq j \leq d. \tag{2.8}$$

A simple special case occurs when $\sigma_2 = 0$. In this case all V^t's are the same, and the optimal weights go to $w^t = \frac{1}{d}$ for all t. In contrast, when σ_2 grows large the weights eventually converge to $w_d = 1$ and $w^t = 0$ for all $t \neq d$. This results in essentially using only y^d to recover X^d and ignoring the rest of the measurements. Combining these, we note that when the σ_2 is small, we can gain by a factor of approximately d over the naïve strategy that ignores dynamics and tries to recover X^d using only y^d. Notice also that the minimum sample complexity is proportional to $\sum_{t=1}^d w_t^2$ when r/d is relatively large. Thus, when σ_2 is small, the required number of measurements can be reduced by a factor of d compared to what would be required to recover X^d using only y^d.

2.3.2 Matrix completion setting

For the matrix completion problem, we consider the following simple uniform sampling ensemble:

Definition 2.3.5. A linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$ is a uniform sampling ensemble (with replacement) if all sensing matrices A_i are i.i.d. uniformly distributed on the set

$$\mathcal{X} = \left\{ e_j(n_1) e_k^T(n_2), 1 \le j \le n_1, 1 \le k \le n_2 \right\},$$

where $e_j(n)$ are the canonical basis vectors in $\mathbb{R}^n$. We let $p = m_0/(n_1 n_2)$ denote the fraction of sampled entries.

For this observation architecture, our analysis is complicated by the fact that it does not satisfy the matrix RIP. (A quick problematic example is a rank-1 matrix with only one non-zero entry.) To handle this we follow the typical approach and restrict our focus to matrices that satisfy certain *incoherence* properties.

Definition 2.3.6. (Subspace incoherence [55]) Let $U \in \mathbb{R}^{n \times r}$ be the orthonormal basis for an r-dimensional subspace $\mathcal{U}$, then the incoherence of $\mathcal{U}$ is defined as $\mu(\mathcal{U}) := \max_{i \in [n]} \frac{\sqrt{n}}{\sqrt{r}} \left\| e_i^T U \right\|_2$, where e_i denotes the i^{th} standard basis vector. We also simply denote $\mu(\text{span}(U))$ as $\mu(U)$.

Definition 2.3.7. (Matrix incoherence [57]) A rank-r matrix $X \in \mathbb{R}^{n_1 \times n_2}$ with SVD $X = U \Sigma V^T$ is incoherent with parameter μ if

$$\left\| U_{:i} \right\|_2 \le \frac{\mu \sqrt{r}}{\sqrt{n_1}} \quad \text{for any } i \in [n_1] \quad \text{and} \quad \left\| V_{:j} \right\|_2 \le \frac{\mu \sqrt{r}}{\sqrt{n_2}} \quad \text{for any } j \in [n_2],$$

i.e., the subspaces spanned by the columns of U and V are both μ-incoherent.

The incoherence assumption guarantees that X is far from sparse, which make it possible to recover X from incomplete measurements since a measurement contains roughly the

same amount of information for all dimensions.

To proceed we also assume that the matrix X^d has "bounded spikiness" in that the maximum entry of X^d is bounded by a, i.e.,$\left\|X^d\right\|_\infty \leq a$. To exploit the spikiness constraint below we replace the optimization constraints $\mathbb{C}(r)$ in (2.3) with $\mathbb{C}(r,a) := \{X \in \mathbb{R}^{n_1 \times n_2} : \text{rank}(X) \leq r, \|X\|_\infty \leq a\}$:

$$\hat{X}^d = \underset{X \in \mathbb{C}(r,a)}{\arg\min}\, \mathcal{L}(X) = \underset{X \in \mathbb{C}(r,a)}{\arg\min}\, \frac{1}{2} \sum_{t=1}^{d} w_t \left\|\mathcal{A}^t(X) - y^t\right\|_2^2. \tag{2.9}$$

Note that Proposition 2.3.1 still holds for (2.9).

To obtain an error bound in the matrix completion case, we lower bound the LHS of 2.4 using a restricted convexity argument (see, for example, [61]) and upper bound the RHS using matrix Bernstein inequality. The result of this approach is the following theorem.

Theorem 2.3.8. *Suppose that we are given measurements as in* (2.1) *where all $\mathcal{A}^t$'s are uniform sampling ensembles. Assume that X^t evolves according to* (2.2)*, has rank r, and is incoherent with parameter μ_0 and $\left\|X^d\right\|_\infty \leq a$. Further assume that the perturbation noise and the measurement noise satisfy the same assumptions in Theorem 2.3.4. If*

$$m_0 \geq D_2 n_{\min} \log^2(n_1+n_2)\phi'(w), \tag{2.10}$$

where $\phi'(w) = \frac{\max_t w_t^2\left((d-t)\mu_0^2 r\sigma_2^2/n_1+\sigma_1^2\right)}{\sum_{t=1}^d w_t^2\left((d-t)\sigma_2^2+\sigma_1^2\right)}$, then the estimator $\hat{X}^d$ from (2.9) *satisfies*

$$\left\|\Delta^d\right\|_F^2 \leq \max\left\{ B_1 := C_2 a^2 n_1 n_2 \sqrt{\frac{\sum_{t=1}^d w_t^2 \log(n_1+n_2)}{m_0}}, B_2 \right\}, \tag{2.11}$$

with probability at least $P_1 = 1 - 5/(n_1+n_2) - 5dn_{\max}\exp(-n_{\min})$, where

$$B_2 = \frac{C_3 r n_1^2 n_2^2 \log(n_1+n_2)}{n_{\min} m_0} \left(\left(\sum_{t=1}^{d} w_t^2\sigma_1^2 + \sum_{t=1}^{d-1}(d-t)w_t^2\sigma_2^2 \right) + \sum_{t=1}^{d} w_t^2 a^2 \right), \tag{2.12}$$

and C_2, C_3, D_2 are absolute positive constants.

If we choose the weights as $w_d = 1$ and $w_t = 0$ for $1 \leq t \leq d-1$, the bound in Theorem 2.3.8 reduces to a result comparable to classical (static) matrix completion results (see, for example, [51] Theorem 7). Moreover, from the B_2 term in (2.11), we obtain the same dependence on m as that of (2.6), i.e., $1/m$. However, there are also a few key differences between Theorem 2.3.4 and our results for matrix completion. In general the bound is loose in several aspects compared to the matrix sensing bound. For example, when m_0 is small, B_1 actually dominates, in which case the dependence on m is actually $1/\sqrt{m}$ instead of $1/m$. When m_0 is sufficiently large, then B_2 dominates, in which case we can consider two cases. The first case corresponds to when a is relatively large compared to σ_1, σ_2 – i.e., the low-rank matrix is spiky. In this case the term containing a^2 in B_2 dominates, and the optimal weights are equal weights of $1/d$. This occurs because the term involving a dominates and there is little improvement to be obtained by exploiting temporal dynamics. In the second case, when a is relatively small compared to σ_1, σ_2 (which is usually the case in practice), the bound can be simplified to

$$\|\Delta\|_F^2 \leq \frac{c_3 r n_1^2 n_2^2 \log(n_1+n_2)}{n_{\min} m_0} \left(\left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right) \right).$$

The above bound is much more similar to the bound in (2.6) from Theorem 2.3.4. In fact, we can also obtain the optimal weights by solving the same quadratic program as (2.7).

When $n_1 \approx n_2$, the sample complexity is $\Theta(n_{\min} \log^2(n_1+n_2)\phi'(w))$. In this case Theorem 2.3.8 also implies a similar sample complexity reduction as we observed in the matrix sensing setting. However, the precise relations between sample complexity and weights w_t's are different in these two cases (deriving from the fact that the proof uses matrix Bernstein inequalities in the matrix completion setting rather than concentration inequalities of Chi-squared variables as in the matrix sensing setting).

2.4 A projected gradient descent algorithm

We first introduce some new variables and notations. For a rank-r matrix X, let the singular value decomposition (SVD) be $X = \bar{U}\Lambda\bar{V}$, where $\bar{U} \in \mathbb{R}^{n_1\times r}, \bar{V} \in \mathbb{R}^{n_2\times r}$ are orthonormal matrices, and $\Lambda \in \mathbb{R}^{r\times r}$ is a diagonal matrix, whose entries are sorted nonzero singular values $\lambda_1(X), \lambda_2(X), \ldots, \lambda_r(X)$. Besides, Let $\tilde{U} = \bar{U}\Lambda^{1/2}$ and $\tilde{V} = \bar{V}\Lambda^{1/2}$, then following [62] and [63], we lift the low-rank matrix X to a positive semidefinite matrix $Y \in \mathbb{R}^{(n_1+n_2)\times(n_1+n_2)}$ as follows:

$$Y = \begin{bmatrix} \tilde{U}\tilde{U}^T & \tilde{U}\tilde{V}^T \\ \tilde{V}\tilde{U}^T & \tilde{V}\tilde{V}^T \end{bmatrix} = ZZ^T,$$

where $Z := [\tilde{U}; \tilde{V}] \in \mathbb{R}^{(n_1+n_2)\times r}$. Note that the above factorization is not unique, so it is convenient to define an class of matrices equivalent to Z:

$$\mathcal{Z} = \left\{ Z' \in \mathbb{R}^{(n_1+n_2)\times r} | Z' = ZR \quad \text{for some } R \in \mathbb{Q}_r \right\},$$

where $\mathbb{Q}_r$ is the set of r-by-r orthonormal matrices. Besides, for convenience let Z_U and Z_V denote the top d_1 and bottom d_2 rows of Z respectively.

Definition 2.4.1. Define the pseudo-metric as the minimal Frobenius norm between Z and Z' with respect to the optimal rotation, namely

$$d(Z, Z') = \min_{\bar{Z}\in\mathcal{Z}} \|Z - Z'\|_F = \min_{R\in\mathbb{Q}_r} \|Z - Z'R\|_F .$$

Though the above pseudo-metric is not exactly the sames as the generalization error metric $\|X - X'\|_F$ in the original matrix space $\mathbb{R}^{n_1\times n_2}$, however the following lemmas state the relationship between them.

Lemma 2.4.2 (Lemma D.1 in [62])**.** *For any $Z, Z' \in \mathbb{R}^{(n_1+n_2)\times r}$ satisfying $d(Z, Z') \leq$*

$\|Z'\|_2/4$ *we have*

$$\left\|Z_U Z_V^T - Z'_U Z'^T_V\right\|_F \leq \frac{9}{4}\lambda_1(Z')d(Z,Z').$$

Lemma 2.4.3 (Lemma 5.14 in [62])**.** *Let $X, X' \in \mathbb{R}^{n_1\times n_2}$ be two rank-r matrices. Assume $\|X - X'\| \leq \frac{1}{2}\lambda_r(X')$, then the following inequality holds*

$$d^2(Z,Z') \leq \frac{2}{\sqrt{2}-1}\frac{\|X-X'\|_F^2}{\lambda_r(X')}.$$

Definition 2.4.4. Define the ball around Z with radius ρ as

$$\mathbb{B}(R;Z) = \left\{Z' \in \mathbb{R}^{(n_1+n_2)\times r} | d(Z',Z) \leq \rho\right\}.$$

In addition to lifting to semidefinite matrix, we also introduce a regularizer g from [62] such that:

$$g(U,V) := \left\|U^TU - V^TV\right\|_F^2.$$

The regularizer is introduced to prove convergence of some non-square matrix estimation problem. The new regularized objective function is

$$\mathcal{F}(U,V) = \mathcal{L}(UV^T) + \frac{1}{8}g(U,V).$$

Now we are ready to present the vanilla projected gradient descent (PGD) algorithm we aim to analyze. The gradient descent updates take the form

$$\begin{bmatrix} U_{t+1} \\ V_{t+1} \end{bmatrix} = \begin{bmatrix} U_t - \eta_t\nabla_U\mathcal{F}(U_t,(V_t)) \\ V_t - \eta_t\nabla_V\mathcal{F}(U_t,(V_t)) \end{bmatrix},$$

where η_t is the step size. Note that for matrix completion, we need an additional step after each gradient step, which projecting the rank-r matrix UV^T to the convex set $\mathbb{C}(r,a)$.

Now we are ready to present the one-step global convergence result for the vanilla PGD

on $\mathcal{F}(U, V)$ in both matrix sensing and matrix completion settings.

Theorem 2.4.5 (Matrix sensing)**.** *Under the setting of Theorem 2.3.4, if sample complexity satisfies* (2.5) *and*

$$\lambda_r^2(X^d) \geq D_3 n_{\max} r \left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right), \tag{2.13}$$

where D_3 is some positive constant. Then there exist positive constants c_1, c_2, c_3, c_4 and c_5, such that with step size $\eta \leq c_1/\lambda_1$ and initial solution Z_0 within $\mathbb{B}(c_2\sqrt{\lambda_r}; Z^d)$, the estimator Z_t produced at iteration $t+1$ of PGD satisfies

$$d^2(Z_{t+1}, Z^d) \leq \left(1 - \frac{2\lambda_r \eta}{45}\right) d^2(Z_t, Z^d) + c_5 \eta r n_{\max} \left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right),$$

with probability at least $1 - dc_3 \exp(-c_4 n_{\max})$.

Remark 2.4.6. Let $\eta = \Theta(1/\lambda_1)$. According to Lemma 2.4.2, as $t \to \infty$ the ultimate statistical error yields

$$\begin{aligned} \left\| X_\infty - X^d \right\|_F^2 &\lesssim r n_{\max} \left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right) \\ &\lesssim \lambda_r^2(X^d). \end{aligned}$$

This is exactly the same as the bound in Theorem 2.3.4.

Remark 2.4.7. Inequality (2.13) implies that to guarantee the one-step convergence of PGD, sufficient energy (in terms of $\lambda_r^2(X^d)$) of the unknown matrix X^d is required.

Theorem 2.4.8 (Matrix completion)**.** *Under the setting of Theorem 2.3.8, if the sample complexity satisfies* (2.10) *and*

$$\lambda_r^2(X^d) \geq D_4 \frac{r m_0 \log(n_1 + n_2)}{n_{\min}} \left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right), \tag{2.14}$$

where D_4 are some positive constant. Then there exist positive constants c_1, c_2 and c_3, such that with step size $\eta \leq c_1\sigma_1$ and initial solution Z_0 satisfies $Z_0 \in \mathbb{B}(c_2\sqrt{\lambda_r}; Z^d)$. the estimator Z_t produced at iteration $t+1$ of PGD satisfies

$$d^2(Z_{t+1}, Z^d) \leq \left(1 - \frac{4\lambda_r\eta}{45}\right) d^2(Z_t, Z^d) + c_3 \max\{B_1, B_2\},$$

with probability at least $P_2 = 1 - 5/(n_1+n_2) - 5dn_{\max}\exp(-n_{\min})$, where

$$B_1 = \max\left\{ n_1 n_2 \sqrt{\frac{\sum_{t=1}^d w_t^2 \log(n_1+n_2)}{m_0}}, \frac{\sum_{t=1}^d w_t^2 \log(n_1+n_2) r n_1^2 n_2^2}{n_{\min} m_0} \right\} \frac{a^2}{\lambda_r}$$

and

$$B_2 = \frac{r m_0 \log(n_1+n_2)}{\eta n_{\min}} \left(\sum_{t=1}^d w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right).$$

Remark 2.4.9. The first part of the bound B_1 comes from the spikiness assumption on X^d. If X^d is not spiky and let $\eta = \Theta(1/\lambda_1)$, then according to Lemma 2.4.2, as $t \to \infty$ the ultimate statistical error yields

$$\begin{aligned} \left\|X_\infty - X^d\right\|_F^2 &\lesssim \frac{r m_0 \log(n_1+n_2)}{n_{\min}} \left(\sum_{t=1}^d w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right) \\ &\lesssim \lambda_r^2(X^d). \end{aligned}$$

This is similar as the second part of the bound in Theorem 2.3.8. Note that the second part of the bound in Theorem 2.3.8 is

$$\left\|\hat{X} - X^d\right\|_F^2 \lesssim \frac{n_1^2 n_2^2}{m_0^2} \frac{r m_0 \log(n_1+n_2)}{n_{\min}} \left(\sum_{t=1}^d w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1} (d-t) w_t^2 \sigma_2^2 \right).$$

The above difference is due to the use of AM-GM inequality in the proof of Theorem 2.3.8.

Remark 2.4.10. Inequality (2.14) implies that to guarantee the one-step convergence of PGD, sufficient energy (in terms of $\lambda_r^2(X^d)$) of the unknown matrix X^d is required. The difference

between (2.14) and (2.13) is due to normalization of the sensing matrices in matrix sensing case.

2.5 Simulations and experiments

2.5.1 Synthetic simulations

Our synthetic simulations consider both matrix sensing and matrix completion, but with an emphasis on matrix completion. We set $n_1 = 100$, $n_2 = 50$, $d = 4$ and $r = 5$. We consider two baselines: **baseline one** is only using y^d to recover X^d and simply ignoring $y^1, \dots y^{d-1}$; **baseline two** is using $\{y^t\}_{t=1}^d$ with equal weights. Note that both of these can be viewed as special cases of LOWEMS with weights $(0, \dots, 0, 1)$ and $(\frac{1}{d}, \frac{1}{d}, \dots, \frac{1}{d})$ respectively. Recalling the formula for the optimal choice of weights in (2.8), it is easy to show that baseline one is equivalent to the case where $\kappa = (\sigma_2^2)/(\sigma_1^2) \to \infty$ and the baseline two equivalent to the case where $\kappa \to 0$. This also makes intuitive sense since $\kappa \to \infty$ means the perturbation is arbitrarily large between time steps, while $\kappa \to 0$ reduces to the static setting.

1). Recovery error. In this simulation, we set $m_0 = 4000$ and set the measurement noise level σ_1 to 0.05. We vary the perturbation noise level σ_2. For every pair of (σ_1, σ_2) we perform 10 trials, and show the average relative recovery error $\left\|\Delta^d\right\|_F^2 / \left\|X^d\right\|_F^2$. Figure 2.1 illustrates how LOWEMS reduces the recovery error compared to our baselines. As one can see, when σ_2 is small, the optimal κ, i.e., σ_2^2/σ_1^2, generates nearly equal weights (baseline two), reducing recovery error approximately by a factor of 4 over baseline one, which is roughly equal to d as expected. As σ_2 grows, the recovery error of baseline two will increase dramatically due to the perturbation noise. However in this case the optimal κ of LOWEMS grows with it, leading to a more uneven weighting and to somewhat diminished performance gains. We also note that, as expected, LOWEMS converges to baseline one when σ_2 is large.

2). Sample complexity. In the interest of conciseness we only provide results here for the matrix completion setting (matrix sensing yields broadly similar results). In this

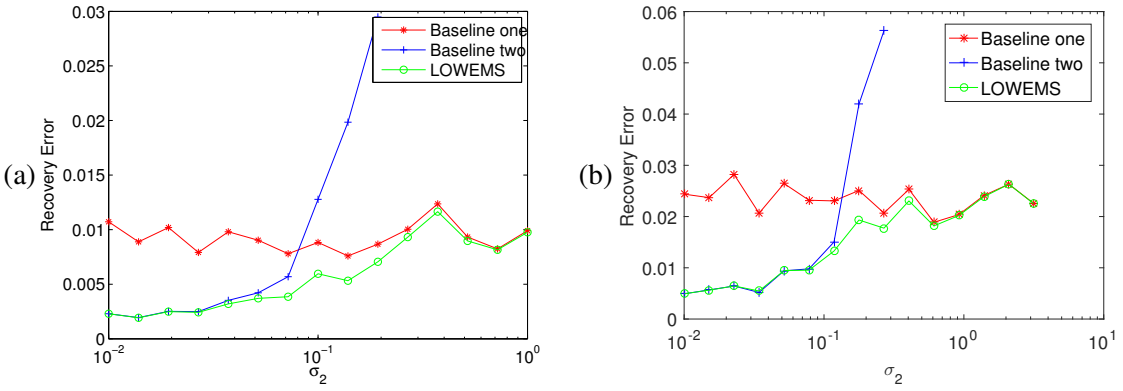

Figure 2.1: Recovery error under different levels of perturbation noise. (a) matrix sensing. (b) matrix completion.

simulation we vary the fraction of observed entries p to empirically find the minimum sample complexity required to guarantee successful recovery (defined as a relative error ≤ 0.08). We compare the sample complexity of the proposed LOWEMS to baseline one and baseline two under different perturbation noise level σ_2 (σ_1 is set as 0.02). For a certain σ_2, the relative recovery error is the averaged over 10 trials. Figure 2.2 illustrates how LOWEMS reduces the sample complexity required to guarantee successful recovery. When the perturbation noise is weaker than the measurement noise, the sample complexity can be reduced approximately by a factor of d compared to baseline one. When the perturbation noise is much stronger than measurement noise, the recovery error of baseline two will increase due to the perturbation noise and hence the sample complexity increase rapidly. However in this case proposed LOWEMS still achieves relatively small sample complexity and its sample complexity converges to baseline one when σ_2 is relatively large.

2.5.2 Real world experiments

We next test the LOWEMS approach in the context of a recommendation system using the (truncated) Netflix dataset. We eliminate those movies with few ratings, and those users rating few movies, and generate a truncated dataset with 3199 users, 1042 movies, 2462840 ratings, and hence the fraction of visible entries in the rating matrix is ≈ 0.74. All the

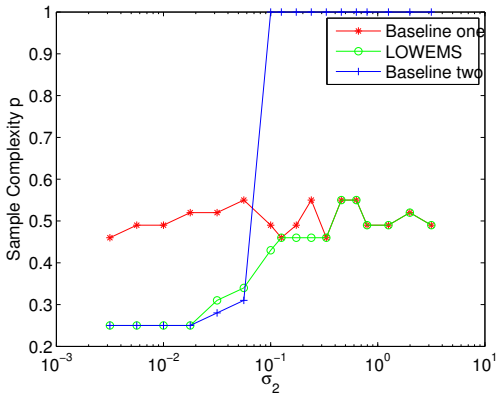

Figure 2.2: Sample complexity under different levels of perturbation noise (matrix completion).

ratings are distributed over a period of 2191 days. For the sake of robustness, we additionally impose a Frobenius norm penalty on the factor matrices U and V. We keep the latest (in time) 10% of the ratings as a testing set. The remaining ratings are split into a validation set and a training set for the purpose of cross validation. We divide the remaining ratings into $d \in \{1, 3, 6, 8\}$ bins respectively with same time period according to their timestamps. We use 5-fold cross validation, and we keep $1/5$ of the ratings from the d^{th} bin as a validation set. The number of latent factors r is set to 10. The Frobenius norm regularization parameter γ is set to 1. We also note that in practice one likely has no prior information on σ_1, σ_2 and hence κ. However, we use model selection techniques like cross validation to select the best κ incorporating the unknown prior information on measurement/perturbation noise. We use root mean squared error (RMSE) to measure prediction accuracy. Since alternating minimization uses a random initialization, we generate 10 test RMSE's (using a boxplot) for the same testing set. Figure 2.3(a) shows that the proposed LOWEMS estimator improves the testing RMSE significantly with appropriate κ. Additionally, the performance improvement increases as d gets larger.

To further investigate how the parameter κ affects accuracy, we also show the validation RMSE compared to κ in Figure 2.3(b). When $\kappa \approx 1$, LOWEMS achieves the best RMSE on the validation data. This further demonstrates that imposing an appropriate dynamic

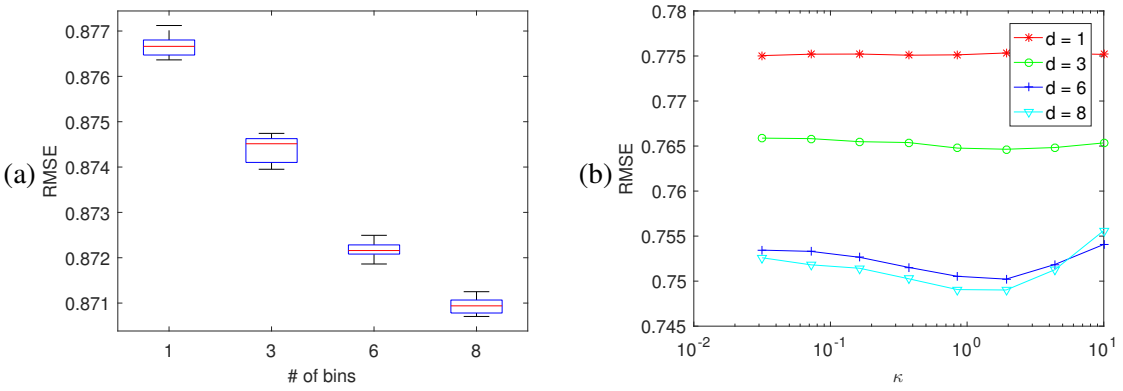

Figure 2.3: Experimental results on truncated Netflix dataset. (a) Testing RMSE vs. number of time steps. (b) Validation RMSE vs. κ.

constraint should improve recovery accuracy in practice.

2.6 Conclusions

In this chapter, we considered the low-rank matrix recovery problem in a novel setting, where one of the factor matrices changes over time. We assume a random walk dynamics model governing the time-evolving factor matrix. We proposed the locally weighted matrix smoothing (LOWEMS) framework, and have established error bounds and convergence guarantee for LOWEMS in both the matrix sensing and matrix completion cases. Our analysis quantifies how the proposed estimator improves recovery accuracy and reduces sample complexity compared to static recovery methods. Finally, we have presented both synthetic and real-world experimental results to verify our analysis and demonstrate superior empirical performance when exploiting dynamics constraints in a recommendation system.

There are several potential remaining improvements. First, it might be possible to conduct a global optimization landscape analysis similar to [64]. This will eliminate the initial solution condition of our global convergence results. Second, similar error bounds and convergence guarantees can be achieved for more general dynamics models, such as a linear dynamical system with a known transition matrix. However, it is worth pointing out that finding similar recovery guarantees for a linear dynamical system with an unknown

transition matrix is still an open problem.

2.7 Technical proof details

2.7.1 Proof of Proposition 2.3.1

Proof. Let $x := \text{vec}(X) \in \mathbb{R}^{n_1 n_2}$ and $\tilde{\mathcal{L}}(x) := \mathcal{L}(X)$. Since the objective function is continuous in X and the set $\mathbb{C}(r)$ is compact, $\mathcal{L}(X)$ achieves a minimizer at some point $\hat{X}^d \in \mathbb{C}(r)$.

Since $\hat{X}^d$ is a minimizer of the constrained problem, then for all matrices $X \in \mathbb{C}(r)$ we have the following inequality

$$\tilde{\mathcal{L}}\left(\hat{x}^d\right) - \tilde{\mathcal{L}}(x) \leq 0. \tag{2.15}$$

By the second-order Taylor's theorem, we expand $\tilde{\mathcal{L}}(x)$ around $x^d = \text{vec}\left(X^d\right)$

$$\tilde{\mathcal{L}}(x) = \tilde{\mathcal{L}}\left(x^d\right) + \left\langle \nabla\tilde{\mathcal{L}}\left(x^d\right), x - x^d \right\rangle + \frac{1}{2}\left\langle \nabla^2\tilde{\mathcal{L}}(\bar{x})\left(x - x^d\right), x - x^d \right\rangle, \tag{2.16}$$

where $\bar{x} = \alpha x^d + (1-\alpha)x$ for some $\alpha \in [0,1]$. Plugging (2.16) with $x = \hat{x}^d$ into (2.15) we obtain

$$\left\langle \nabla\tilde{\mathcal{L}}\left(x^d\right), \hat{x}^d - x^d \right\rangle + \frac{1}{2}\left\langle \nabla^2\tilde{\mathcal{L}}(\bar{x})\left(\hat{x}^d - x^d\right), \hat{x}^d - x^d \right\rangle \leq 0. \tag{2.17}$$

Through some algebraic manipulation we have the following expression for the gradient of $\tilde{\mathcal{L}}(x)$:

$$\nabla\tilde{\mathcal{L}}(x) = \text{vec}\left(\sum_{t=1}^{d} w_t \mathcal{A}^{t*}\left[\mathcal{A}^t(X) - y^t\right]\right). \tag{2.18}$$

Based on the above gradient it follows that

$$\nabla^2\tilde{\mathcal{L}}(x)\, b = \text{vec}\left(\sum_{t=1}^{d} w_t \mathcal{A}^{t*}\left[\mathcal{A}^t(B)\right]\right), \tag{2.19}$$

where $b = \text{vec}(B)$.

Now based on (2.18) and (2.19), the absolute value of first term in (2.17) can be bounded as

$$\begin{aligned}\left|\langle\nabla\tilde{\mathcal{L}}\left(x^d\right),\hat{x}^d-x^d\rangle\right| &= \left|\left\langle\sum_{t=1}^{d}w_t\mathcal{A}^{t*}\left[\mathcal{A}^t\left(X^d\right)-y^t\right],\Delta^d\right\rangle\right| \\ &\leq \left\|\sum_{t=1}^{d}w_t\mathcal{A}^{t*}\left[\mathcal{A}^t\left(X^d\right)-y^t\right]\right\|_2\left\|\Delta^d\right\|_* \\ &\leq \left\|\sum_{t=1}^{d}w_t\mathcal{A}^{t*}\left(h^t-z^t\right)\right\|_2\sqrt{2r}\left\|\Delta^d\right\|_F\end{aligned} \tag{2.20}$$

The first inequality above used the trace dual norm inequality, while the second inequality follows from a basic inequality for rank-$2r$ matrices. Similarly the second term in (2.17) is

$$\begin{aligned}\frac{1}{2}\left\langle\nabla^2\tilde{\mathcal{L}}\left(\bar{x}\right)\left(\hat{x}^d-x^d\right),\hat{x}^d-x^d\right\rangle &= \frac{1}{2}\left\langle\sum_{t=1}^{d}w_t\mathcal{A}^{t*}\mathcal{A}^t\left(\Delta^d\right),\Delta^d\right\rangle \\ &= \frac{1}{2}\sum_{t=1}^{d}w_t\left\langle\mathcal{A}^t\left(\Delta^d\right),\mathcal{A}^t\left(\Delta^d\right)\right\rangle.\end{aligned} \tag{2.21}$$

The result follows from combining (2.20) and (2.21). Note that the above proof holds if we replace $\mathbb{C}\left(r,\right)$ with $\mathbb{C}\left(r,a\right)$, which completes our proof. □

2.7.2 Proof of Theorem 2.3.4

Proof. The proof consists of lower bounding the LHS of (2.4) and upper bounding the RHS of (2.4).

We use the following lemma to lower bound $\sum_{t=1}^{d}w_t\left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2$.

Lemma 2.7.1. *Suppose the linear operator $\mathcal{A}^t:\mathbb{R}^{n_1\times n_2}\to\mathbb{R}^{m_0}$ is random Gaussian ensemble for all $1\leq t\leq d$. If $m_0>Dn_{\max}r\sum_{t=1}^{d}w_t^2$, the composite operator $\left\{\sqrt{w_t}\mathcal{A}^t\right\}_{t=1}^{d}$ satisfies the rank-$2r$ matrix RIP with constant $\delta_{2r}\leq\delta$ with probability exceeding $1-C\exp\left(-cm_0\right)$, where D,C and c (which depends on σ) are absolute positive constants.*

Proof. See Section 2.7.3. □

Next lemma gives us an upper bound for the stochastic error $\left\|\sum_{t=1}^{d} w_t \mathcal{A}^{t*}\left(h^t - z^t\right)\right\|_2$.

Lemma 2.7.2. *Under the assumptions of Theorem 2.3.4, when $m_0 \geq Dn_{\max}$, we have*

$$\left\|\sum_{t=1}^{d} w_t \mathcal{A}^{t*}\left(h^t - z^t\right)\right\|_2 \leq C_1 \sqrt{n_{\max}(1+\delta_1)\left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d-1}(d-t) w_t^2 \frac{2rn_2}{m_0}\sigma_2^2\right)}$$

with probability exceeding $1 - dC\exp(-cn_2)$, where D, C_1, C, c are positive constants and δ_1 is the rank-1 matrix RIP parameter for all $\mathcal{A}^t$'s.

Proof. See Section 2.7.4. □

Theorem 2.3.4 follows by combining Lemma 2.7.1, Lemma 2.7.2 and Definition 2.3.3.

□

2.7.3 Proof of Lemma 2.7.1

Proof. First we introduce the following theorem providing a double-sided tail bound on the sum of independent sub-exponential random variables.

Theorem 2.7.3. *For independent X_i sub-exponential with parameters (σ_i, b_i), with mean μ_i,*

$$\mathbb{P}\left(\left|\sum_{i=1}^{n}(X_i - \mu_i)\right| \geq nt\right) \leq 2\exp\left(-\frac{nt^2}{2\left(\sigma^2 + bt\right)}\right),$$

where $\sigma^2 = \sum_i \sigma_i^2$ and $b = \max_i b_i$.

We now lower bound $\sum_{t=1}^{d} w_t \left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2$. Since all $\mathcal{A}^t$'s are Gaussian random measurement ensembles, then a particular measurement $\left\langle A_i^t, \Delta^d\right\rangle^2$ is distributed as $m_0^{-1}\left\|\Delta^d\right\|_F^2 \chi^2(1)$. Therefore $\sum_{t=1}^{d} w_t \left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2 = \sum_{t,i} w_t \left\langle A_i^t, \left(\Delta^d\right)\right\rangle^2$ is a weighted sum of i.i.d. $\chi^2(1)$ random variables. Since $\chi^2(1)$ is sub-exponential with parameters $(4,4)$, Theorem 2.7.3 implies a double-sided tail bound for $\sum_{t=1}^{d} w_t \left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2$: for any given $\Delta^d \in \mathbb{R}^{n_1 \times n_2}$ and

any fixed $0 < s < 1$

$$\mathbb{P}\left(\left|\sum_{t=1}^{d} w_t \left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2 - \left\|\Delta^d\right\|_F^2\right| \leq s\left\|\Delta^d\right\|_F^2\right) \leq 2\exp\left(-\frac{m_0 s^2}{8\sum_{t=1}^{d} w_t^2 + 8w_{\max}s}\right),$$

where $w_{\max} = \max\{w_1, \ldots, w_d\}$. The probability can be further simplified if s is very small ($\leq 1/d$).

Rank of Δ^d is at most $2r$ since $\hat{X}^d, X^d$ are rank-r matrices. By Theorem 2.3 in [31] (one may see the proof if necessary) if $m_0 > Dn_{\max} r \sum_{t=1}^{d} w_t^2$, the composite operator $\left\{\sqrt{w_t}\mathcal{A}^t\right\}_{t=1}^{d}$ satisfies the rank-$2r$ matrix RIP with constant $\delta_{2r} \leq \delta$ with probability exceeding $1 - C\exp\left(-cm_0\right)$, where C and c (depends on δ) are absolute positive constants.

□

2.7.4 Proof of Lemma 2.7.2

Proof. Let $W = \sum_{t=1}^{d} w_t \mathcal{A}^{t*}\left(h^t - z^t\right)$ and $n = n_{\max}$ for short. Following the basic framework of the proof of Lemma 1.1 in [31], we use ϵ-nets method to bound the stochastic error $\|W\|_2$. The operator norm of W is

$$\|W\|_2 = \sup_{\|u\|=\|v\|=1} \langle u, Wv\rangle,$$

Consider a $1/4$-net $\mathcal{N}_{1/4}$ of the unite sphere S^{n-1} with $\left|\mathcal{N}_{1/4}\right| \leq 12^n$ (see (III.1) in [31]). For any $v, u \in S^{n-1}$

$$\begin{aligned}\langle u, Wv\rangle &= \langle u - u_0, Wv\rangle + \langle u_0, W\left(v - v_0\right)\rangle + \langle u_0, Wv_0\rangle \\ &\leq \|W\|_2 \|u - u_0\|_2 + \|W\|_2 \|v - v_0\|_2 + \langle u_0, Wv_0\rangle,\end{aligned}$$

for some $v_0, w_0 \in \mathcal{N}_{1/4}$ obeying $\|u - u_0\|_2 \leq 1/4$ and $\|v - v_0\| \leq 1/4$. So the operator norm of W is

$$\|W\|_2 \leq 2 \sup_{u_0, v_0 \in \mathcal{N}_{1/4}} \langle u_0, Wv_0\rangle.$$

For fixed u_0, v_0

$$\langle u_0, Wv_0\rangle = \text{Tr}\left(u_0^T W v_0\right) = \text{Tr}\left(v_0 u_0^T W\right) = \left\langle u_0 v_0^T, W\right\rangle = \sum_{t=1}^{d} w_t \left\langle \mathcal{A}^t\left(u_0 v_0^T\right), h^t - z^t\right\rangle.$$

Let $Z = \sum_{t=1}^{d} w_t \left\langle \mathcal{A}^t\left(u_0 v_0^T\right), z^t\right\rangle$ and $H = \sum_{t=1}^{d} w_t \left\langle \mathcal{A}^t\left(u_0 v_0^T\right), h^t\right\rangle$. Since for all $1 \leq t \leq d$, entries of z^t are i.i.d. $\mathcal{N}\left(0, \sigma_1^2\right)$, therefore $Z \sim \mathcal{N}\left(0, \sigma_Z^2\right)$, where the variance σ_Z^2 is

$$\sigma_Z^2 = \sum_{t=1}^{d} w_t^2 \left\|\mathcal{A}^t\left(u_0 v_0^T\right)\right\|_2^2 \sigma_1^2 \leq \sum_{t=1}^{d} w_t^2 \left(1+\delta_1\right) \left\|u_0 v_0^T\right\|_F^2 \sigma_1^2 = \sum_{t=1}^{d} w_t^2 \left(1+\delta_1\right) \sigma_1^2. \tag{2.22}$$

The first inequality uses the matrix RIP for rank-1 matrices. For a fixed t, $\mathcal{A}^t$ satisfies the rank-1 matrix RIP with constant δ_1, with probability at least $1 - C_2 \exp(-c_2 m_0)$ provided that $m_0 \geq D_2 n$ by Theorem 2.3 in [31], where C_2, c_2 and D_2 are fixed positive constants. Then by a union bound, for all $1 \leq t \leq d$, $\mathcal{A}^t$ satisfies the rank-1 matrix RIP property with parameter σ_1, with probability at least $1 - dC_2 \exp(-c_2 m_0)$ provided that $m_0 \geq D_2 n$.

We now simplify H as

$$\begin{aligned}
H = \sum_{t=1}^{d} w_t \left\langle \mathcal{A}^t\left(u_0 v_0^T\right), h^t\right\rangle &= \sum_{t=1}^{d-1} w_t \left\langle \mathcal{A}^t\left(u_0 v_0^T\right), \sum_{s=t+1}^{d} \mathcal{A}^t\left[U\left(\epsilon^s\right)^T\right]\right\rangle \\
&= \sum_{s=2}^{d} \sum_{t=1}^{s-1} \left\langle w_t \mathcal{A}^t\left(u_0 v_0^T\right), \mathcal{A}^t\left[U\left(\epsilon^s\right)^T\right]\right\rangle \\
&= \sum_{s=2}^{d} \sum_{t=1}^{s-1} \left\langle w_t \mathcal{A}^{t*}\mathcal{A}^t\left(u_0 v_0^T\right), U\left(\epsilon^s\right)^T\right\rangle \\
&= \sum_{s=2}^{d} \sum_{t=1}^{s-1} \sum_{i=1}^{m_0} \left\langle w_t \left[\mathcal{A}^t\left(u_0 v_0^T\right)\right]_i A_i^t, U\left(\epsilon^s\right)^T\right\rangle \\
&= \sum_{s=2}^{d} \left\langle \sum_{t=1}^{s-1} w_t \left\|\mathcal{A}^t\left(u_0 v_0^T\right)\right\|_2 U^T A^t, \left(\epsilon^s\right)^T\right\rangle,
\end{aligned}$$

where $A^t \in \mathbb{R}^{n_1 \times n_2}$ contains i.i.d. $\mathcal{N}\left(0, 1/m_0\right)$ entries. The last equality uses the property that sum of independent Gaussian variables is also Gaussian, and the variance is the sum of individual variances. Since for all $2 \leq s \leq d$, entries of ϵ^s are i.i.d. $\mathcal{N}\left(0, \sigma_2^2\right)$, therefore

$H \sim \mathcal{N}(0, \sigma_H^2)$, where the variance σ_H^2 is

$$\begin{aligned}
\sigma_H^2 = \sum_{s=2}^{d} \left\| \sum_{t=1}^{s-1} w_t \left\| \mathcal{A}^t \left(u_0 v_0^T \right) \right\|_2 U^T A^t \right\|_F^2 \sigma_2^2 &\overset{(\xi_1)}{\leq} \sum_{s=2}^{d} \left\| \sum_{t=1}^{s-1} w_t \sqrt{1+\delta_1} U^T A^t \right\|_F^2 \sigma_2^2 \\
&\overset{(\xi_2)}{=} \sum_{s=2}^{d} \sum_{t=1}^{s-1} w_t^2 \left(1+\delta_1\right) \left\| U^T B^s \right\|_F^2 \sigma_2^2 \\
&= \sum_{s=2}^{d} \sum_{t=1}^{s-1} w_t^2 \left(1+\delta_1\right) \frac{1}{m_0} \chi_s^2 \left(r n_2\right) \sigma_2^2 \\
&\overset{(\xi_3)}{\leq} \sum_{s=2}^{d} \sum_{t=1}^{s-1} w_t^2 \left(1+\delta_1\right) \frac{1}{m_0} 3 m_0 \sigma_2^2 \\
&= \sum_{t=1}^{d-1} (d-t) w_t^2 \left(1+\delta_1\right) \sigma_2^2.
\end{aligned} \tag{2.23}$$

Inequality (ξ_1) holds with probability exceeding $1 - dC_2 \exp(-c_2 m_0)$ provided that $m_0 \geq Dn$ based on the matrix RIP for rank-1 matrices as used while bounding σ_Z^2. Equality (ξ_2) uses the property that sum of independent Gaussian variables is also Gaussian and entries of B^s are i.i.d. $\mathcal{N}(0, 1/m_0)$. Inequality (ξ_3) holds with probability at least $1 - dC_3 \exp(-c_3 m_0)$ by the concentration property of correlated Chi-squared variables.

Since the measurement noise Z and dynamic perturbation H are independent, then $\langle u_0, W v_0 \rangle \sim \mathcal{N}(0, \sigma_Z^2 + \sigma_H^2)$. Then by a standard tail bound for Gaussian random variables we have

$$\mathbb{P}\left(|\langle u_0, W v_0 \rangle| > \lambda \right) \leq 2 \exp\left(-\frac{\lambda^2}{2\left(\sigma_H^2 + \sigma_Z^2\right)} \right).$$

Therefore by an standard union bound we bound the stochastic error

$$\mathbb{P}\left(\|W\|_2 \geq C_0 \sqrt{n\left(\sigma_H^2 + \sigma_Z^2\right)} \right) \leq 2 \left| \mathcal{N}_{1/4} \right|^2 \exp\left(-\frac{C_0^2 n}{8} \right) \leq 2 \exp\left(-cn \right), \tag{2.24}$$

where $c = \frac{C_0^2}{8} - 2 \log 12$. To ensure $c > 0$, we require $C_0 > 4\sqrt{\log 12}$.

Combining (2.22), (2.23), and (2.24), if $m_0 \geq Dn$ we have

$$\|W\|_2 \leq C_0 \sqrt{n\left((1+\delta_1)\sum_{t=1}^{d} w_t^2\left(\sigma_1^2 + (d-t)\frac{5rn_2}{m_0}\sigma_2^2\right)\right)}$$

with probability exceeding $1 - [dC_2\exp(-c_2 m_0) + dC_3\exp(-c_3 m_0) + 2\exp(-cn)] \geq 1 - dC\exp(-cn_2)$. □

2.7.5 Proof of Theorem 2.3.8

Proof. The proof follows the same framework of the proof of Theorem 7 in [51].

Before we lower bound $\sum_{t=1}^{d} w_t \left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2$, we consider the following constraint set for a given $0 < r \leq n$:

$$\mathcal{E}(r) = \left\{ X \in \mathbb{C}(r) : \|X\|_\infty = 1, \|X\|_F^2 \geq n_1 n_2 \sqrt{\frac{2048\sum_{t=1}^{d} w_t^2 \log(n_1+n_2)}{\log(6/5) m_0}} \right\}.$$

Define the following random matrix

$$\Sigma_R = \sum_{t=1}^{d}\sum_{i=1}^{m_0} w_t \gamma_i^t A_i^t,$$

where γ_i^t is Rademacher variable.

The following lemma bounds the restricted strong convexity (see [61]) of the operator $\left\{\sqrt{w_t}\mathcal{A}^t\right\}_{t=1}^{d}$.

Lemma 2.7.4. *Suppose all $\mathcal{A}^t$'s are fixed uniform sampling ensembles. For all $X \in \mathcal{E}(r)$*

$$\sum_{t=1}^{d} w_t \left\|\mathcal{A}^t(X)\right\|_2^2 \geq \frac{p}{2}\|X\|_F^2 - \frac{44rn_1n_2}{m_0}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2 \tag{2.25}$$

with probability at least $1 - \frac{2}{(n_1+n_2)}$.

Proof. See Section 2.7.6. □

Note that $\left\|\Delta^d\right\|_\infty \leq \left\|\hat{X}^d\right\|_\infty + \left\|X^d\right\|_\infty \leq 2\left\|X^d\right\|_\infty$. To proceed, we consider the following two cases.

Case I. $\frac{\Delta^d}{2\left\|X^d\right\|_\infty} \notin \mathcal{E}(2r)$.

Following the definition of $\mathcal{E}(2r)$ we have

$$\left\|\Delta^d\right\|_F^2 \leq c_2 \left\|X^d\right\|_\infty^2 n_1 n_2 \sqrt{\frac{\sum_{t=1}^d w_t^2 \log(n_1+n_2)}{m_0}},$$

where $C_2 = 4\sqrt{\frac{2048}{\log(6/5)}}$. This yields the first part of inequality (2.11) in Theorem 2.3.8.

Case II. $\frac{\Delta^d}{2\left\|X^d\right\|_\infty} \in \mathcal{E}(2r)$.

Since $\frac{\Delta^d}{2\left\|X^d\right\|_\infty} \in \mathcal{E}(2r)$, applying Lemma 2.7.4 yields

$$\sum_{t=1}^d w_t \left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2 \geq \frac{p}{2}\left\|\Delta^d\right\|_F^2 - \frac{362rn_1n_2}{m_0}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2\left\|X^d\right\|_\infty^2. \tag{2.26}$$

Combining (2.26) and (2.4) yields

$$\begin{aligned}\frac{p}{2}\left\|\Delta^d\right\|_F^2 &\leq 2\sqrt{2r}\left\|\sum_{t=1}^d w_t \mathcal{A}^{t*}\left(h^t - z^t\right)\right\|_2 \left\|\Delta^d\right\|_F + \frac{362rn_1n_2}{m_0}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2\left\|X^d\right\|_\infty^2 \\ &\leq \frac{8r}{p}\left\|\sum_{t=1}^d w_t \mathcal{A}^{t*}\left(h^t - z^t\right)\right\|_2^2 + \frac{p}{4}\left\|\Delta^d\right\|_F^2 + \frac{362rn_1n_2}{m_0}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2\left\|X^d\right\|_\infty^2.\end{aligned}$$

The above inequality can be further simplified as

$$\left\|\Delta^d\right\|_F^2 \leq \frac{32rn_1^2n_2^2}{m_0^2}\left\|\sum_{t=1}^d w_t \mathcal{A}^{t*}\left(h^t - z^t\right)\right\|_2^2 + \frac{1448rn_1^2n_2^2}{m_0^2}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2\left\|X^d\right\|_\infty^2. \tag{2.27}$$

Next we bound $\mathbb{E}(\|\Sigma_R\|)$ in the following lemma.

Lemma 2.7.5. *Suppose all $\mathcal{A}^t$'s are fixed uniform sampling ensembles. For*

$$m_0 \geq Dn_{\min}\log(n_1+n_2)\,\phi(w),$$

where $\phi(w) = \frac{w_{\max}^2}{\sum_{t=1}^d w_t^2}$*, there exists an absolute positive constant* C *such that*

$$\mathbb{E}(\|\Sigma_R\|) \leq C\sqrt{\frac{2e\log(n_1+n_2)\sum_{t=1}^d w_t^2 m_0}{n_{\min}}}. \tag{2.28}$$

The proof is not provided since it is almost the same as that of Lemma 6 in [51] with some minor modifications. Note that our results are a bit stronger compared to Lemma 6 in [51], since we are dealing with bounded variables.

Now we upper bound the stochastic error $\|J\|_2^2 := \left\|\sum_{t=1}^d w_t \mathcal{A}^{t*}\left(h^t - z^t\right)\right\|_2^2$. First, we rewrite J as

$$J = \sum_{t=1}^d w_t \mathcal{A}^{t*}\mathcal{A}^t \left[U\left(\sum_{s=t+1}^d \epsilon^s\right)^T + Z^t\right],$$

where each entry of the random matrix $Z^t \in \mathbb{R}^{n_1\times n_2}$ is i.i.d. Gaussian distributed with variance σ_1^2. Set $Y^t = U\left(\sum_{s=t+1}^d \epsilon^s\right)^T$ and $F^t = Y^t + Z^t$. Note that F^t may be correlated for different $1 \leq t \leq d$, though for a given t the entries of F^t are independent.

We now introduce an $n_1 \times n_2$ random matrix G^t that has exactly one non-zero entry:

$$G^t = w_t n_1 n_2 F_{ij}^t E_{ij}, \quad \text{with probability } \frac{1}{n_1 n_2},$$

where E_{ij} is the canonical basis of matrices with dimension $n_1 \times n_2$. We also introduce the following random matrix H^t, which is the average of m_0 independent copies of G^t:

$$H^t = \frac{1}{m_0}\sum_{i=1}^{m_0} G_i^t \quad \text{where each } G_i^t \text{ is an independent copy of } G^t.$$

Then J can be decomposed as sum of independent random matrices: $J = \frac{m_0}{n_1 n_2}\sum_{t=1}^d H^t$. It is immediate that

$$\mathbb{E}G^t = \mathbb{E}H^t = w_t F^t, \quad \mathbb{E}J = \frac{m_0}{n_1 n_2}\sum_{t=1}^d w_t F^t.$$

Before we proceed we introduce a lemma describing the spectral norm deviation of a sum of uncentered random matrices from its mean value.

Lemma 2.7.6. *(Corollary 6.1.2 in [65]) Consider a finite sequence $\{S_k\}$ of independent random matrices with common dimension $n_1 \times n_2$. Assume that each matrix has uniformly bounded deviation from its mean:*

$$\|S_k - \mathbb{E}S_k\| \leq L \quad \textit{for each index } k.$$

Consider the sum

$$Z = \sum_k S_k.$$

Let $\rho(Z)$ denotes the matrix variance statistic of the sum:

$$\begin{aligned}\rho(Z) &= \max\left\{\left\|\mathbb{E}[(Z-\mathbb{E}Z)(Z-\mathbb{E}Z)^T]\right\|, \left\|\mathbb{E}[(Z-\mathbb{E}Z)^T(Z-\mathbb{E}Z)]\right\|\right\} \\ &= \max\left\{\left\|\sum_k \mathbb{E}[(S_k-\mathbb{E}S_k)(S_k-\mathbb{E}S_k)^T]\right\|, \left\|\sum_k \mathbb{E}[(S_k-\mathbb{E}S_k)^T(S_k-\mathbb{E}S_k)]\right\|\right\}.\end{aligned}$$

Then for all $s \geq 0$,

$$\mathbb{P}\left(\|Z-\mathbb{E}Z\| \geq s\right) \leq (n_1+n_2)\exp\left(\frac{-s^2/2}{\rho(Z)+Ls/3}\right).$$

We are going to apply the above uncentered Bernstein inequality to the sum of dm_0 independent random matrices $\sum_{t=1}^{d} H^t = \frac{1}{m_0}\sum_{t=1}^{d}\sum_{k=1}^{m_0} G_k^t$. Before doing so, we note that for given t and k,

$$\left\|G_k^t - \mathbb{E}G_k^t\right\| \leq \left\|G_k^t\right\| + \left\|\mathbb{E}G_k^t\right\| \leq \left\|G_k^t\right\| + \mathbb{E}\left\|G_k^t\right\| \leq 2\left\|G_k^t\right\|.$$

The first inequality uses the triangle inequality; the second is Jensen's inequality.

To control $\rho(\sum_{t=1}^{d} H^t)$, first note that

$$\begin{aligned}\mathbf{0} \preceq \sum_t \sum_k \mathbb{E}\left[G_k^t - \mathbb{E}G_k^t)(G_k^t - \mathbb{E}G_k^t)^T\right] &= \sum_t \sum_k \mathbb{E}\left[(G_k^t(G_k^t)^T\right] - (\mathbb{E}G_k^t)(\mathbb{E}G_k^t)^T \\ &\preceq \sum_t \sum_k \mathbb{E}\left[G_k^t(G_k^t)^T\right] \\ &= m_0 \sum_t \mathbb{E}\left[G^t(G^t)^T\right].\end{aligned}$$

The third relation holds because $(\mathbb{E}G_k^t)(\mathbb{E}G_k^t)^T$ is positive semidefinite; the last relation uses the fact that for a fixed t, G_k^t are random matrices following identical distributions independently for all $1 \le k \le m_0$. Now we can control $\rho(\sum_{t=1}^{d} H^t)$ in the following

$$\rho\left(\sum_{t=1}^{d} H^t\right) \le \frac{1}{m_0} \max\left\{\left\|\sum_t \mathbb{E}\left[(G^t(G^t)^T\right]\right\|, \left\|\sum_t \mathbb{E}\left[(G^t)^T G^t\right]\right\|\right\}.$$

Set $\rho_0 := \max\left\{\left\|\sum_{t=1}^{d} \mathbb{E}(G^t(G^t)^T)\right\|, \left\|\sum_{t=1}^{d} \mathbb{E}((G^t)^T G^t)\right\|\right\}$. Then the remaining work is to uniformly upper bound $\|G_k^t\|$ for all $1 \le t \le d$ and $1 \le k \le m_0$ and upper bound ρ_0.

First we turn to the uniform bound on the spectral norm of the random matrix G_k^t for all $1 \le t \le d$ and $1 \le k \le m_0$. We have for all $1 \le t \le d$ and $1 \le k \le m_0$

$$\left\|G_k^t\right\| \le \max_{i,j,t} w_t \left\|n_1 n_2 F_{ij}^t E_{ij}\right\| = n_1 n_2 \max_{i,j,t} w_t |F_{ij}^t|.$$

Since $\mu(U) \le \mu_0$, the variance of each entry of the random matrix F^t can be bounded as $\mathrm{Var}(F_{ij}^t) \le \frac{\mu_0^2 r}{n_1}\sigma_2^2(d-t) + \sigma_1^2$. Let $\sigma_{\max}^2 = \max_t w_t^2 \left(\frac{\mu_0^2 r}{n_1}\sigma_2^2(d-t) + \sigma_1^2\right)$. Then by the tail probability of Gaussian random variables and the standard union bound (over i, j), for all $1 \le t \le d$ and $1 \le k \le m_0$ we have

$$\mathbb{P}\left(\left\|G_k^t\right\| \le n_1 n_2 \sqrt{2\log(d(n_1+n_2)n_1 n_2)\sigma_{\max}^2} =: L\right) \ge 1 - 2/(n_1+n_2).$$

Second we turn to the computation of $\mathbb{E}(G^t(G^t)^T)$. We have

$$\mathbb{E}(G^t(G^t)^T) = w_t^2 n_1^2 n_2^2 \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}(F_{ij}^t)^2 E_{ij}E_{ij}^T \frac{1}{n_1 n_2} = w_t^2 n_1 n_2 \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}(F_{ij}^t)^2 E_{ii}.$$

Similarly $\mathbb{E}((G^t)^T G^t) = w_t^2 n_1 n_2 \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}(F_{ij}^t)^2 E_{jj}$. Then

$$\rho = n_1 n_2 \max\left\{\max_i \sum_{t=1}^{d}\sum_{j=1}^{n_2} w_t^2 (F_{ij}^t)^2, \max_j \sum_{t=1}^{d}\sum_{i=1}^{n_1} w_t^2 (F_{ij}^t)^2\right\}.$$

Let $a_i = \sum_{t=1}^{d}\sum_{j=1}^{n_2} w_t^2 (F_{ij}^t)^2$ and $b_j = \sum_{t=1}^{d}\sum_{i=1}^{n_1} w_t^2 (F_{ij}^t)^2$. We first bound $\max_i a_i$. Note that $a_i = \sum_{t=1}^{d} w_t^2 \sum_{j=1}^{n_2}(Y_{ij}^t + Z_{ij}^t)^2 \le 2\sum_{t=1}^{d} w_t^2 \sum_{j=1}^{n_2}[(Y_{ij}^t)^2 + (Z_{ij}^t)^2]$. Note that for $1 \le i \le n_1$ and $1 \le t \le d$, $\sum_{j=1}^{n_2}(Z_{ij}^t)^2 \sim \sigma_1^2 \chi^2(n_2)$ and are independent. So by the tail bound of Chi-squared variable and the standard union bound (over i and t) we have

$$\mathbb{P}\left(\max_i \sum_{t=1}^{d} w_t^2 \sum_{j=1}^{n_2}(Z_{ij}^t)^2 \le 5n_2 \sum_{t=1}^{d} w_t^2 \sigma_1^2\right) \ge 1 - dn_1 \exp(-n_2). \tag{2.29}$$

Similarly we have

$$\mathbb{P}\left(\max_j \sum_{t=1}^{d} w_t^2 \sum_{i=1}^{n_2}(Z_{ij}^t)^2 \le 5n_1 \sum_{t=1}^{d} w_t^2 \sigma_1^2\right) \ge 1 - dn_2 \exp(-n_1). \tag{2.30}$$

For $\sum_{j=1}^{n_2}(Y_{ij}^t)^2$, note that Y_{ij}^t is Gaussian distributed and the variance is not greater than $\frac{\mu_0^2 r}{n_1}(d-t)\sigma_2^2$ for all i, j, t, since $\mu(U) \le \mu_0$. For a fixed i, for all $1 \le j \le n_2$, Y_{ij}^t are independent Gaussian random variables. So given i and t, applying the tail bound of Chi-squared random variables yields

$$\mathbb{P}\left(\sum_{j=1}^{n_2}(Y_{ij}^t)^2 \le 5n_2(d-t)\frac{\mu_0^2 r}{n_1}\sigma_2^2\right) \ge 1 - \exp(-n_2).$$

By the standard union bound (over i and t) we have

$$\mathbb{P}\left(\max_i \sum_{t=1}^{d} w_t^2 \sum_{j=1}^{n_2} (Y_{ij}^t)^2 \le 5n_2 \frac{\mu_0^2 r}{n_1} \sum_{t=1}^{d} (d-t) w_t^2 \sigma_2^2\right) \ge 1 - dn_1 \exp(-n_2). \quad (2.31)$$

Now we turn to $\sum_{i=1}^{n_1} (Y_{ij}^t)^2$, which follows a Chi-squared distribution $(d-t)\sigma_2^2 \chi^2(r)$, since

$$\sum_{i=1}^{n_1} (Y_{ij}^t)^2 = (Y_{:j}^t)^T Y_{:j}^t = \bar{\epsilon}_{j:}^t U^T U \left(\bar{\epsilon}_{j:}^t\right)^T = \bar{\epsilon}_{j:}^t \left(\bar{\epsilon}_{j:}^t\right)^T$$

where $\bar{\epsilon}^t = \sum_{s=t+1}^{d} \epsilon^s$. The last equality uses the fact that U is orthonormal. Then by the tail bound of Chi-squared random variables and the standard union bound (over j and t) we have

$$\mathbb{P}\left(\max_j \sum_{t=1}^{d} w_t^2 \sum_{i=1}^{n_1} (Y_{ij}^t)^2 \le 5n_1 \sum_{t=1}^{d} (d-t) w_t^2 \sigma_2^2\right) \ge 1 - dn_2 \exp(-n_1). \quad (2.32)$$

Combining (2.29) and (2.31) yields

$$\mathbb{P}\left(\max_i a_i \le 10n_2 \sum_{t=1}^{d} w_t^2 \left(\sigma_1^2 + \frac{\mu_0^2 r}{n_1}(d-t)\sigma_2^2\right)\right) \ge 1 - 2dn_1 \exp(-n_2). \quad (2.33)$$

Similarly combining (2.30) and (2.32) yields

$$\mathbb{P}\left(\max_j b_j \le 10n_1 \sum_{t=1}^{d} w_t^2 \left(\sigma_1^2 + (d-t)\sigma_2^2\right)\right) \ge 1 - 2dn_2 \exp(-n_1). \quad (2.34)$$

Note that $1 \le \mu_0 \le \sqrt{n_1}/\sqrt{r}$, so $\frac{\mu_0^2 r}{n_1} \le 1$. Now we are ready to bound ρ_0 by combining (2.33) and (2.34):

$$\mathbb{P}\left(\rho_0 \le 10 n_{\max} n_1 n_2 \left(\sum_{t=1}^{d} w_t^2 \sigma_1^2 + \sum_{t=1}^{d} w_t^2 (d-t)\sigma_2^2\right) =: \nu\right) \ge 1 - 4dn_{\max} \exp(-n_{\min}). \quad (2.35)$$

Now by Lemma 2.7.6, we have

$$\mathbb{P}\left(\left\|\sum_{t=1}^{d} H^t - \sum_{t=1}^{d} w_t F^t\right\| \geq s\right) \leq (n_1+n_2)\exp\left(\frac{-m_0 s^2/2}{\nu + 2Ls/3}\right).$$

If we let $s = \sqrt{\frac{8\log(n_1+n_2)\nu}{m_0}}$ and substitute this into the above matrix Bernstein inequality we obtain

$$\mathbb{P}\left(\left\|\sum_{t=1}^{d} H^t - \sum_{t=1}^{d} w_t F^t\right\| \geq \sqrt{\frac{8\log(n_1+n_2)\nu}{m_0}}\right) \leq 1/(n_1+n_2).$$

A hidden condition when the above inequality holds is that ν dominates the denominator of the exponential term. The remaining work is to have sufficiently large m_0 to guarantee that ν dominates the denominator of the exponential, which follows

$$\nu \geq 2/3L\sqrt{\frac{8\log(n_1+n_2)\nu}{m_0}}.$$

The above inequality immediately implies that

$$m_0 \geq \frac{32}{45} n_{\min} \log(d(n_1+n_2)n_1n_2)\log(n_1+n_2)\frac{\max_t w_t^2\left((d-t)\frac{\mu_0^2 r}{n_1}\sigma_2^2 + \sigma_1^2\right)}{\sum_{t=1}^{d} w_t^2\left((d-t)\sigma_2^2+\sigma_1^2\right)}.$$

Note that $n_1+n_2 > n_i, i=1,2$, and $n_1+n_2 > d$, then the above sample complexity can be simplified as

$$m_0 \geq \frac{128}{45} n_{\min} \log^2(n_1+n_2)\frac{\max_t w_t^2\left((d-t)\frac{\mu_0^2 r}{n_1}\sigma_2^2 + \sigma_1^2\right)}{\sum_{t=1}^{d} w_t^2\left((d-t)\sigma_2^2+\sigma_1^2\right)}. \tag{2.36}$$

The remaining work is to bound $\left\|\sum_{t=1}^{d} w_t F^t\right\|$. First we note that each entry of F^t is Gaussian and the variance is not greater than $\sigma_1^2 + (d-t)\sigma_2^2$. Then, according to results on

bounds for the spectral norm of i.i.d. Gaussian ensemble, we have

$$\mathbb{P}\left(\left\|\sum_{t=1}^{d} w_t F^t\right\| \leq 2\sqrt{\sum_{t=1}^{d} w_t^2\left(\sigma_1^2+(d-t)\sigma_2^2\right)}\sqrt{n_{\max}}\right) \geq 1-C_1\exp(-c_2 n_{\max}), \quad (2.37)$$

where C_1, c_2 are absolute positive constants. Note that $C_1\exp(-c_2 n_{\max}) \ll dn_{\max}\exp(-n_{\min})$.

Now we are ready to bound $\|J\|_2^2$. With probability at least $1-\frac{3}{n_1+n_2}-5dn_{\max}\exp(-n_{\min})$ we have

$$\begin{aligned}
\|J\|_2^2 &\leq p^2\left(\left\|\sum_{t=1}^{d} w_t F^t\right\| + \sqrt{\frac{8\log(n_1+n_2)\nu}{m_0}}\right)^2 \\
&\leq 320p^2\max\{n_1n_2\log(n_1+n_2)/m_0, 1\}n_{\max}\sum_{t=1}^{d} w_t^2((d-t)\sigma_2^2+\sigma_1^2) \\
&= 320p^2\sum_{t=1}^{d} w_t^2((d-t)w_2^2+\sigma_1^2)n_1n_2\log(n_1+n_2)n_{\max}/m_0 \\
&= \frac{320m_0\log(n_1+n_2)\sum_{t=1}^{d} w_t^2((d-t)\sigma_2^2+\sigma_1^2)}{n_{\min}}.
\end{aligned} \quad (2.38)$$

The first equality uses the fact that $m_0 < n_1n_2\log(n_1+n_2)$.

Combining (2.27),(2.28) and (2.38) yields the second part of inequality (2.11) in Theorem 2.3.8. □

2.7.6 Proof of Lemma 2.7.4

Proof. The proof is almost the same as the proof of Lemma 12 in [51] with some minor modifications.

Set $\mathcal{F} = \frac{44rn_1n_2}{m_0}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2$. We will show that the probability of the following bad event is small:

$$\mathcal{B} = \left\{\exists X \in \mathcal{E}(r) \text{ such that } \left|\sum_{t=1}^{d} w_t\left\|\mathcal{A}^t(X)\right\|_2^2 - p\|X\|_F^2\right| > \frac{p}{2}\|X\|_F^2 + \mathcal{F}\right\}.$$

Note that $\mathcal{B}$ contains the complement of the event in Lemma 2.7.4.

We use a peeling argument to bound the probability of $\mathcal{B}$. Let $\nu = \sqrt{\frac{2048\sum_{t=1}^{d} w_t^2 \log(n_1+n_2)}{\log(6/5)m_0}}$ and $\alpha = 6/5$. For $l \in \mathcal{N}$ let

$$S_l = \left\{ X \in \mathcal{E}(r) : \nu\alpha^{l-1} \leq \frac{1}{n_1 n_2} \|X\|_F^2 \leq \nu\alpha^l \right\}.$$

Then if event $\mathcal{B}$ holds for some $X \in \mathcal{E}(r)$, it must be that X belongs to some S_l and

$$\left| \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(X) \right\|_2^2 - p\|X\|_F^2 \right| > \frac{p}{2}\|X\|_F^2 + \mathcal{F} > \frac{5}{12}\alpha^l \nu m_0 + \mathcal{F}. \tag{2.39}$$

For $T > \nu$ consider the set

$$\mathcal{E}(r,T) = \left\{ X \in \mathcal{E}(r) : \|X\|_F^2 \leq n_1 n_2 T \right\}$$

and the event

$$\mathcal{B}_l = \left\{ \exists X \in \mathcal{E}(r, \alpha^l\nu) \text{ such that } \left| \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(X) \right\|_2^2 - p\|X\|_F^2 \right| > \frac{5}{12}\alpha^l\nu m_0 + \mathcal{F} \right\}. \tag{2.40}$$

Note that $X \in S_l$ implies that $X \in \mathcal{E}(r, \alpha^l\nu)$. Then (2.39) implies that $\mathcal{B}_l$ holds and $\mathcal{B} \subset \cup\mathcal{B}_l$. Thus, it is sufficient to bound the probability of the simpler event $\mathcal{B}_l$ and then apply the union bound. Such a bound is given by the following lemma. Its proof is given in Section 2.7.7. Let

$$H_T = \sup_{X \in \mathcal{E}(r,T)} \left| \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(X) \right\|_2^2 - p\|X\|_F^2 \right|.$$

Lemma 2.7.7. *Suppose all $\mathcal{A}^t$'s are fixed uniform sampling ensembles. Then*

$$\mathbb{P}\left(H_T > \frac{5}{12}\alpha^l\nu m_0 + \mathcal{F} \right) \leq \exp\left(\frac{-c_5 m_0 T^2}{\sum_{t=1}^{d} w_t^2} \right),$$

where $c_5 = 1/4096$.

The above lemma implies that $\mathbb{P}(\mathcal{B}_l) \leq \exp(-c_5 m_0 \alpha^{2l} \nu^2)$. By a union bound, we have

$$\mathbb{P}(\mathcal{B}) \leq \sum_{l=1}^{\infty} \mathbb{P}(\mathcal{B}_l) \leq \sum_{l=1}^{\infty} \exp\left(\frac{-c_5 m_0 \alpha^{2l} \nu^2}{\sum_{t=1}^{d} w_t^2}\right) \leq \sum_{l=1}^{\infty} \exp\left(\frac{-(2c_5 m_0 \log(\alpha) \nu^2) l}{\sum_{t=1}^{d} w_t^2}\right),$$

where the last inequality uses the bound $e^x \geq x$. Substituting $v = \sqrt{\frac{2048 \sum_{t=1}^{d} w_t^2 \log(n_1+n_2)}{\log(6/5) m_0}}$ into the above summation we obtain

$$\mathbb{P}(\mathcal{B}) \leq 2/(n_1 + n_2).$$

This completes the proof. □

2.7.7 Proof of Lemma 2.7.7

Proof. The proof is almost the same as the proof of Lemma 14 in [51] with some minor modifications.

By Massart's concentration inequality (see, e.g., [66], Theorem 14.2), we have

$$\mathbb{P}\left(H_T \geq \mathbb{E}(H_T) + \frac{1}{9}\frac{5}{12} m_0 T\right) \leq \exp\left(\frac{-c_5 m_0 T^2}{\sum_{t=1}^{d} w_t^2}\right), \tag{2.41}$$

where $c_5 = 1/4096$. Next we bound the expectation $\mathbb{E}(H_T)$. Using a symmetrization argument we obtain

$$\mathbb{E}(H_T) \leq 2\mathbb{E}\left(\sup_{X \in \mathcal{E}(r,T)} \left| \sum_{t=1}^{d} w_t \gamma_i^t \sum_{i=1}^{m_0} \left\langle A_i^t, X \right\rangle^2 \right|\right),$$

where γ_i^t is a Rademacher variable (independent on both i and t). The assumption $\|X\|_\infty = 1$ implies that $|\langle A_i^t, X \rangle| \leq 1$. Then the contraction inequality yields

$$\mathbb{E}(H_T) \leq 8\mathbb{E}\left(\sup_{X \in \mathcal{E}(r,T)} \left| \sum_{t=1}^{d} w_t \gamma_i^t \sum_{i=1}^{m_0} \left\langle A_i^t, X \right\rangle \right|\right) = 8\mathbb{E}\left(\sup_{X \in \mathcal{E}(r,T)} |\langle \Sigma_R, X \rangle|\right),$$

where $\Sigma_R = \sum_{t=1}^{d} \sum_{i=1}^{m_0} w_t \gamma_i^t A_i^t$. Since $X \in \mathcal{E}(r, T)$, we have

$$\|X\|_* \leq \sqrt{r} \|X\|_F \leq \sqrt{r n_1 n_2 T}.$$

Then by the trace duality inequality, we obtain

$$\mathbb{E}(H_T) \leq 8\sqrt{r n_1 n_2 T} \mathbb{E} \|\Sigma_R\|_2 .$$

Finally using

$$\frac{1}{9}\frac{5}{12} m_0 T + 8\sqrt{r n_1 n_2 m_0 T} \frac{1}{\sqrt{m_0}} \mathbb{E} \|\Sigma_R\|_2 \leq \frac{1}{9}\frac{5}{12} m_0 T + \frac{8}{9}\frac{5}{12} m_0 T + \frac{44 r n_1 n_2}{m_0} \left(\mathbb{E} \|\Sigma_R\|_2\right)^2$$

combined with (2.41) we complete the proof. □

2.7.8 Proof of Theorem 2.4.5

Our goal is to prove the stated algorithm converges to X^d with some statistical error under some mild conditions. Before that we define the following expected loss function with respective to observation noise

$$\tilde{\mathcal{L}}(X) = \mathbb{E}\mathcal{L}(X) = \frac{1}{2} \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(X) - \mathcal{A}^t(X^t) \right\|_2^2 .$$

Unlike the expected loss function defined in [67], the matrix we want to estimate X^d is not the global optimum $\tilde{\mathcal{L}}(X)$, i.e., $\nabla \tilde{\mathcal{L}}(X^d) \neq 0$. So we cannot directly apply the results in [67]. However we can still define another expected loss function as follows:

$$\bar{\mathcal{L}}(X) = \frac{1}{2} \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(X) - \mathcal{A}^t(X^d) \right\|_2^2 .$$

One can check that the gradient at X^d vanishes. The above expected loss function is different from those presented in [67] in the sense that our expected loss function accounts for both

observation and perturbation noise.

Before presenting our main convergence analysis, we also introduce several definitions for differentiable functions.

Definition 2.7.8 (Restricted Strongly Convexity)**.** Differentiable function f is restricted strongly convex with parameter μ, such that for any rank-r matrices $X, Y \in \mathbb{R}^{n_1 \times n_2}$

$$f(Y) \geq f(X) + \langle \nabla f(X), Y - X \rangle + \frac{\mu}{2} \|Y - X\|_F^2 .$$

Definition 2.7.9 (Restricted Strongly Smoothness)**.** Differentiable function f is restricted strongly smooth with parameter μ, such that for any rank-r matrices $X, Y \in \mathbb{R}^{n_1 \times n_2}$

$$f(Y) \leq f(X) + \langle \nabla f(X), Y - X \rangle + \frac{L}{2} \|Y - X\|_F^2 .$$

We use the following known results on the non-convex matrix estimation algorithm.

Theorem 2.7.10 (One step convergence ([67]))**.** *Recall that X^* is the unknown rank-r matrix we want to estimate. The expected loss function $\bar{\mathcal{L}}(X)$ satisfies $\bar{\mu}$-restricted strongly convexity and $\bar{L}$-restricted strongly smoothness condition. For any $Z_0 \in \mathbb{B}(c_2\sqrt{\lambda_r}; Z^*)$, where $c_2 \leq \min\{1/4, \sqrt{2\bar{\mu}'/(5(4\bar{L}+1))}\}$, and if the following statistical error bound holds*

$$\left\|\nabla \mathcal{L}(X) - \nabla \bar{\mathcal{L}}(X)\right\|_2^2 \leq \frac{c_2^2 \bar{\mu}' \lambda_r^2}{10 c_3 r}$$

then with step size $\eta = c_1/\lambda_1$, where $c_1 \leq \min\{1/(64\bar{L}), 1/32\}$ and $\bar{\mu}' = \min\{\bar{\mu}, 1\}$, the estimator at iteration t of vanilla PGD satisfies

$$d^2(Z_{t+1}, Z^*) \leq \left(1 - \frac{c_1 \bar{\mu}'}{10\kappa}\right) d^2(Z_t, Z^*) + c_3 \eta r \left\|\nabla \mathcal{L}(X_t) - \nabla \bar{\mathcal{L}}(X_t)\right\|_2^2 ,$$

where $c_3 = 2/\bar{L} + 4/\bar{\mu}$ and κ is the condition number of X^*.

Remark 2.7.11. Theorem 2.7.10 states that if the following conditions are satisfied: 1) The population loss function is restricted strongly convex and smooth; 2) the statistical error is bound, then the vanilla PGD converge globally providing that the initial solution is close enough to the unknown true solution.

Proof of Theorem 2.4.5. First we check the restricted strongly convexity and smoothness conditions of the expected loss function $\bar{\mathcal{L}}(X)$.

Let $X, Y \in \mathbb{R}^{n_1 \times n_2}$ be any two rank-r matrices, then for the expected loss function $\bar{\mathcal{L}}(X)$, we have

$$\bar{\mathcal{L}}(Y) - \bar{\mathcal{L}}(X) - \left\langle \nabla \bar{\mathcal{L}}(X), Y - X \right\rangle = \frac{1}{2} \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(Y - X) \right\|_2^2 .$$

We apply the matrix RIP results in Lemma 2.7.1. Since $m_0 \geq D_3 n_{\max} r \sum_{t=1}^{d} w_t^2$, then with probability exceeding $1 - C_1 \exp(c m_0)$ we have

$$\frac{1}{2}(1 - \delta_{2r}) \left\| Y - X \right\|_F^2 \leq \bar{\mathcal{L}}(Y) - \bar{\mathcal{L}}(Y) - \left\langle \nabla \bar{\mathcal{L}}(X), Y - X \right\rangle \leq \frac{1}{2}(1 + \delta_{2r}) \left\| Y - X \right\|_F^2 ,$$

where c, C_1 are universal constants.

This yields

$$\bar{\mu} = (1 - \delta_{2r}) \quad \text{and} \quad \bar{L} = (1 + \delta_{2r}), \tag{2.42}$$

where δ_{2r} is the matrix RIP parameter depending on D_3.

Now we bound the statistical error term, which is

$$\begin{aligned}
\left\| \nabla \mathcal{L}(X) - \nabla \bar{\mathcal{L}}(X) \right\|_2^2 &= \left\| \sum_{t=1}^{d} w_t \mathcal{A}^{t*} \left[\mathcal{A}^t (X) - y^t \right] - \sum_{t=1}^{d} w_t \mathcal{A}^{t*} \left[\mathcal{A}^t (X) - \mathcal{A}(X^d) \right] \right\|_2^2 \\
&= \left\| \sum_{t=1}^{d} w_t \mathcal{A}^{t*} \left[\mathcal{A}^t \left(X^d \right) - \mathcal{A}^t (X^t) - z^t \right] \right\|_2^2 \\
&= \left\| \sum_{t=1}^{d} w_t \mathcal{A}^{t*} \left(h^t - z^t \right) \right\|_2^2 ,
\end{aligned}$$

Since $m_0 \geq D_1 n_{\max}$, by Lemma 2.7.2 we have

$$\left\|\nabla\mathcal{L}(X) - \nabla\bar{\mathcal{L}}(X)\right\|_2^2 \leq C_2^2 n_{\max}(1+\delta_1)\left(\sum_{t=1}^{d} w_t^2\sigma_1^2 + \sum_{t=1}^{d-1}(d-t)w_t^2\sigma_2^2\right)$$

with probability exceeding $1 - dC_3\exp(C_4 n_{\max})$, where C_2, C_3, C_4 are some positive constants depending on D_3.

Now check the statistical error assumption for Theorem 2.7.10. Let $D_2 = \frac{10C_2^2c_3}{c_2^2\bar{\mu}'}(1+\delta_1)$. Then by (2.13) we have

$$\begin{aligned}\left\|\nabla\mathcal{L}(X) - \nabla\bar{\mathcal{L}}(X)\right\|_2^2 &\leq C_2^2 n_{\max}(1+\delta_1)\left(\sum_{t=1}^{d} w_t^2\sigma_1^2 + \sum_{t=1}^{d-1}(d-t)w_t^2\sigma_2^2\right)\\ &\leq \frac{C_2^2}{D_2 r}\lambda_r^2\\ &\leq \frac{c_2^2\bar{\mu}'\lambda_r^2}{10c_3 r}.\end{aligned}$$

Now all conditions in Theorem 2.7.10 is proven to hold with probability at least $1 - dC_3\exp(C_4 n_{\max})$. By Theorem 2.7.10, we complete the proof.

□

2.7.9 Proof of Theorem 2.4.8

Proof. First we introduce The following lemma, which shows the restricted strong convexity and smoothness (see [51]) of the operator $\left\{\sqrt{w_t}\mathcal{A}^t\right\}_{t=1}^d$.

Lemma 2.7.12. *Suppose all $\mathcal{A}^t$'s are fixed uniform sampling ensembles. For all $X \in \mathcal{E}(r)$*

$$\left|\sum_{t=1}^{d} w_t\left\|\mathcal{A}^t(X)\right\|_2^2 - p\left\|X\right\|_F^2\right| \geq \frac{p}{2}\left\|X\right\|_F^2 + \frac{44rn_1n_2}{m_0}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2 \tag{2.43}$$

with probability at least $1 - \frac{2}{n_1+n_2}$.

Proof. The proof is exactly the same as the proof of Lemma 2.7.4. See Section 2.7.6. □

Let X_t be the solution at step t of the PGD and $\Delta^d = X_t - X^d$. Note that $\Delta^d \in \mathbb{C}(2r, 2a)$. Following a similar proof strategy for Theorem 2.3.8, we consider two cases.

Case I. $\frac{\Delta^d}{2\left\|X^d\right\|_\infty} \notin \mathcal{E}(2r)$.

Following the definition of $\mathcal{E}(2r)$ we have

$$\left\|\Delta^d\right\|_F^2 \leq C_2 \left\|X^d\right\|_\infty^2 n_1 n_2 \sqrt{\frac{\sum_{t=1}^d w_t^2 \log(n_1+n_2)}{m_0}},$$

where $C_2 = 4\sqrt{\frac{2048}{\log(6/5)}}$.

Case II. $\frac{\Delta^d}{2\left\|X^d\right\|_\infty} \in \mathcal{E}(2r)$.

Since $\frac{\Delta^d}{2\left\|X^d\right\|_\infty} \in \mathcal{E}(2r)$, applying Lemma 2.7.12 yields

$$\left|\sum_{t=1}^d w_t \left\|\mathcal{A}^t\left(\Delta^d\right)\right\|_2^2 - p\left\|\Delta^d\right\|_F^2\right| \geq \frac{p}{2}\|\Delta\|_F^2 + \frac{362 r n_1 n_2}{m_0}\left(\mathbb{E}(\|\Sigma_R\|)\right)^2 \left\|X^d\right\|_\infty^2.$$

According to Lemma 2.7.5 we have

$$\mathbb{E}(\|\Sigma_R\|) \leq C\sqrt{\frac{2e\log\left(n_1+n_2\right)\sum_{t=1}^d w_t^2 m_0}{n_{\min}}},$$

where C is some positive constant. Now we consider two cases.

Case II.a. $\left\|X^d\right\|_\infty^2 \geq \frac{m_0 p\left\|\Delta^d\right\|_F^2}{1448 r n_1 n_2 \left(\mathbb{E}(\|\Sigma_R\|)\right)^2}$

In this case we have

$$\begin{aligned}\left\|\Delta^d\right\|_F^2 &\leq \frac{1448 r n_1 n_2 \left(\mathbb{E}(\|\Sigma_R\|)\right)^2}{m_0 p}\left\|X^d\right\|_\infty^2 \\ &\leq \frac{2896 C^2 e \log(n_1+n_2)\sum_{t=1}^d w_t^2 r n_1^2 n_2^2}{m_0 n_{\min}}\left\|X^d\right\|_\infty^2.\end{aligned}$$

Now by Lemma 2.4.3 we can conclude that

$$d^2\left(Z_t, Z^d\right) \lesssim \max\left\{n_1 n_2\sqrt{\frac{\sum_{t=1}^d w_t^2\log(n_1+n_2)}{m_0}}, \frac{\sum_{t=1}^d w_t^2 \log(n_1+n_2) r n_1^2 n_2^2}{n_{\min} m_0}\right\}\frac{a^2}{\lambda_r}$$

This is the B_1 part in the bound of the Theorem 2.4.8.

Case II.b. $\left\|X^d\right\|_\infty^2 \leq \frac{m_0 p \left\|\Delta^d\right\|_F^2}{1448 r n_1 n_2 (\mathbb{E}(\|\Sigma_R\|))^2}$

Then it is obvious that

$$\frac{362 r n_1 n_2}{m_0} \left(\mathbb{E}(\|\Sigma_R\|)\right)^2 \left\|X^d\right\|_\infty^2 \leq \frac{p}{4} \left\|\Delta^d\right\|_F^2 .$$

So for the function $\bar{\mathcal{L}}(X)$, the restricted convexity constant is $\bar{\mu} = 3/4$ and the restricted smoothness constant is $\bar{L} = 5/4$.

Now check the statistical error assumption for Theorem 2.7.10. Let $D_4 = \frac{3200 c_3}{c_2^2 \bar{\mu}'}$ and $J = \sum_{t=1}^d w_t \mathcal{A}^{t*} \left(h^t - z^t\right)$. By the proof of Theorem 2.3.8 in Section 2.7.5 and (2.14) we have

$$\begin{aligned} \|J\|_2^2 &\leq \frac{320 m_0 \log(n_1 + n_2) \sum_{t=1}^d w_t^2 ((d-t)\sigma_2^2 + \sigma_1^2)}{n_{\min}} \\ &\leq \frac{c_2^2 \bar{\mu}'}{10 c_3 r} \lambda_r^2. \end{aligned}$$

Now all conditions in Theorem 2.7.10 are proven to hold with high probability. Applying Theorem 2.7.10, we complete the proof. □

CHAPTER 3
ONE-BIT LOW-RANK MATRIX SMOOTHING AND FASTER SIMULTANEOUS RECOVERY

.

In the last chapter, we proposed the LOWEMS framework as an approach to dynamic matrix recovery, and we obtained recovery guarantees under the random walk dynamics model. In this chapter, we consider two practical extensions for LOWEMS. First, we consider a new setting in which we aim to recover an underlying dynamically-evolving low-rank matrix from binary observations. This problem arises in a variety of applications, such as personalized learning and tweet recommendation. We propose the one-bit LOWEMS approach and test it in the context of personalized learning. In the second part of this chapter, we solve the problem of simultaneously recovering a series of low-rank matrices based on the LOWEMS framework. We propose a simultaneously LOWEMS (S-LOWEMS) estimator. Our synthetic simulations and real-world experiments show that, compared to the original LOWEMS estimator, the proposed S-LOWEMS estimator not only recovers a series of low-rank matrices with a small computational overhead but also improves the recovery accuracy and reduces the sample complexity.

3.1 One-bit measurement

3.1.1 Introduction

Although low-rank models have been used in many application, in some contexts a linear observation model is not appropriate. For example in the context of personalized learning systems (see [70]), we may only have access to binary responses (right/wrong) for the

Material in this section is joint work with Mark Davenport and has led to publications [68, 69]

students' answers to the assigned questions from which we hope to learn. Since a student's knowledge/skill changes (and hopefully improve) throughout the learning process as a result of lectures, homeworks, and so on, our goal is to unite the recent work in the area of one-bit matrix completion [50, 71, 72] with recent efforts in the context of dynamic matrix completion, including [46].

3.1.2 Problem formulation

We assume that we only have one-bit observations on a subset of the entries at each time-step, i.e., we observe

$$Y_{i,j}^t = \begin{cases} +1 & \text{with prob. } f\left(X_{i,j}^t\right), \\ -1 & \text{with prob. } 1 - f\left(X_{i,j}^t\right) \end{cases} \quad \text{for } (i,j) \in \Omega^t, \tag{3.1}$$

where f is fixed and known. Two common choices for f are the logistic function $f(x) = 1/(1+e^{-x/\sigma_1})$ and the probit function $f(x) = \Phi(x/\sigma_1)$, where $\Phi(x)$ is the cumulative distribution function of standard Gaussian and σ_1^2 is the variance of zero-mean logistic (Gaussian) distribution. We also denote $p^t = |\Omega^t|/(n_1 n_2)$.

The negative log-likelihood for the given problem at time t is

$$\mathcal{L}(X; \Omega^t, Y^t) = -\sum_{(i,j)\in\Omega^t} \left\{ \mathbb{I}_{Y_{i,j}^t=1} \log(f(X_{i,j})) + \mathbb{I}_{Y_{i,j}^t=-1} \log(1 - f(X_{i,j})) \right\}. \tag{3.2}$$

The proposed one-bit LOWEMS (Locally Weighted Matrix Smoothing) is formulated as the following optimization program:

$$\hat{X}^d = \underset{X\in\mathbb{C}(r,\alpha)}{\arg\min}\, \mathcal{F}(X) = \underset{X\in\mathbb{C}(r,\alpha)}{\arg\min} \sum_{t=1}^{d} w_t \mathcal{L}(X; \Omega^t, Y^t), \tag{3.3}$$

where $\mathcal{C}(r,\alpha) := \{X \in \mathbb{R}^{n_1\times n_2} : \text{rank}(X) \le r, \|X\|_\infty \le \alpha\}$ and $\{w_t\}_{t=1}^d$ are non-negative weights. The optimal weights can be computed as (2.8).

The program in (3.3) can be reformulated as

$$\hat{X}^d = \underset{X=UV^T, \|X\|_\infty \leq \alpha}{\arg\min} \mathcal{F}\left(UV^T\right), \tag{3.4}$$

where $U \in \mathbb{R}^{n_1 \times r}, V \in \mathbb{R}^{n_2 \times r}$. We use alternating gradient descent to minimize $\mathcal{F}(U, V)$, which alternatively applies a gradient descent step over U (or V) while holding V (or U) fixed until a stopping criterion is reached. Our choice of stepsize is the safe-guard LBB (long Barzilai-Borwein) stepsize [73]. We also rescale U and V following the gradient descent step so that $\left\|UV^T\right\|_\infty \leq \alpha$ is satisfied at each step.

3.1.3 Simulations and experiments

We set $n_1 = 100$, $n_2 = 50$, $d = 4$, $r = 2$, $p^t = 0.8$ for all t, and use the logistic function for f. We consider two baselines: **baseline one** is only using y^d to recover X^d and simply ignoring $y^1, \ldots y^{d-1}$; **baseline two** is using $\{y^t\}_{t=1}^d$ with equal weights. Note that both of these can be viewed as special cases of one-bit LOWEMS with weights $(0, \ldots, 0, 1)$ and $(\frac{1}{d}, \frac{1}{d}, \ldots, \frac{1}{d})$ respectively.

Figure 3.1 shows that the recovery performance is poor when noise is either too large or too small, a similar phenomenon as observed in [50]. Figure 3.2 illustrates that one-bit LOWEMS reduces the recovery error compared to our baselines, which is also observed in the continuous observation setting [46]. Figure 3.3 shows that one-bit LOWEMS reduces the sample complexity required to guarantee successful recovery (defined as a relative error ≤ 0.4).

Furthermore, we test the one-bit LOWEMS approach in the context of personalized learning using the *ASSISTment* dataset (for a precise description, see [74]). We truncate the dataset by eliminating students/questions with less than 100 responses. We keep a portion (10%) of the most recent data as the testing set, and use the remaining data to learn the matrix. To exploit the dynamic constraint, we divide the training set into d bins

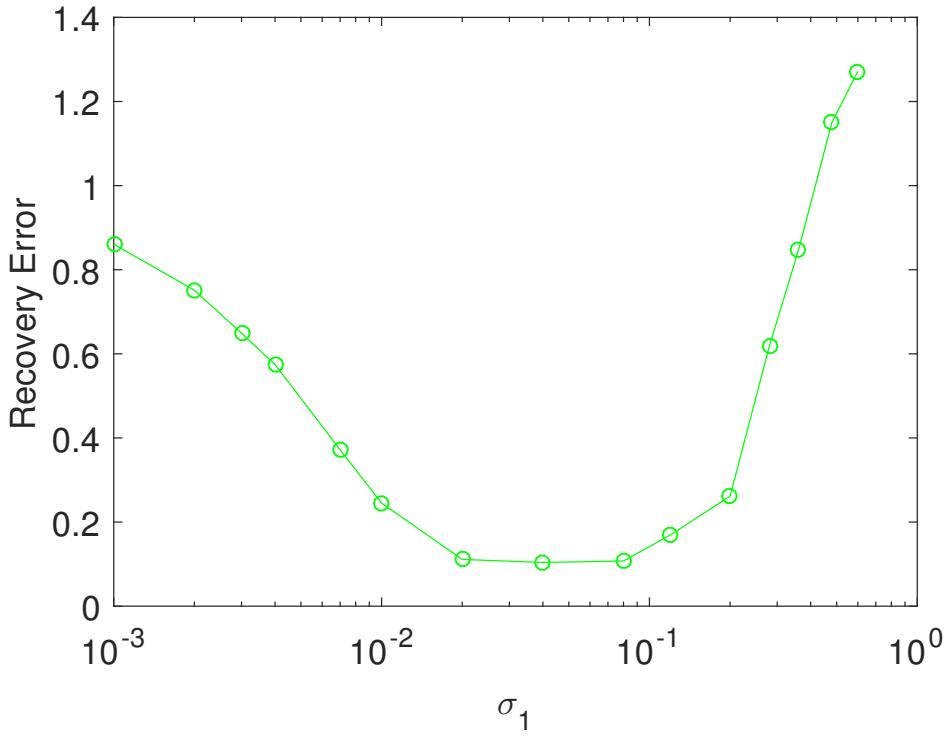

Figure 3.1: Recovery error vs. observation noise ($\sigma_2 = 0.1$).

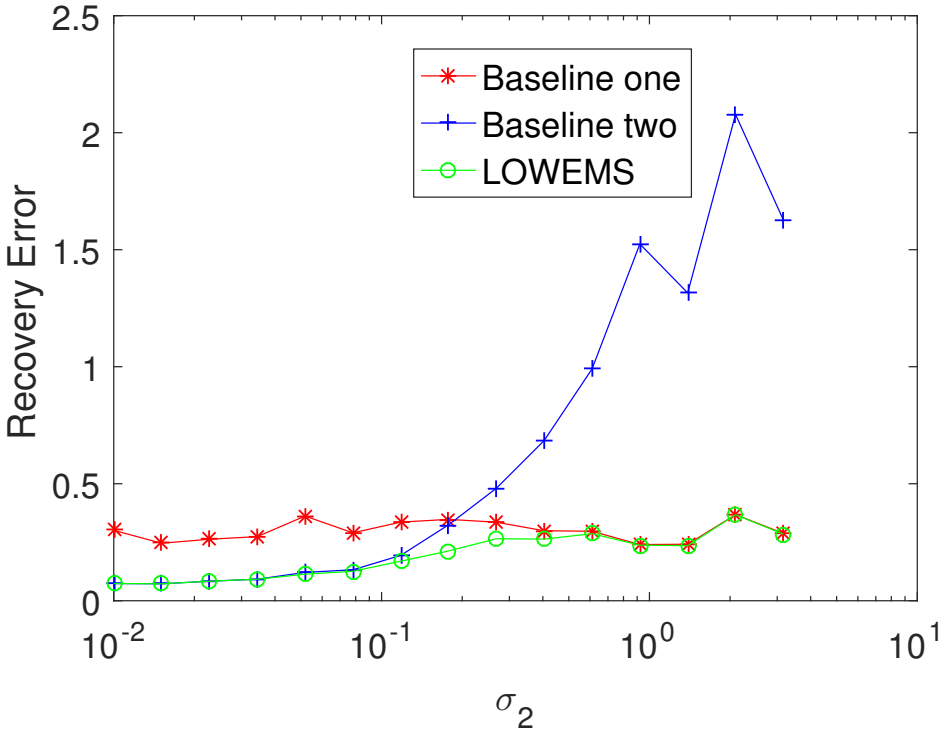

Figure 3.2: Recovery error vs. perturbation noise ($\sigma_1 = 0.1$).

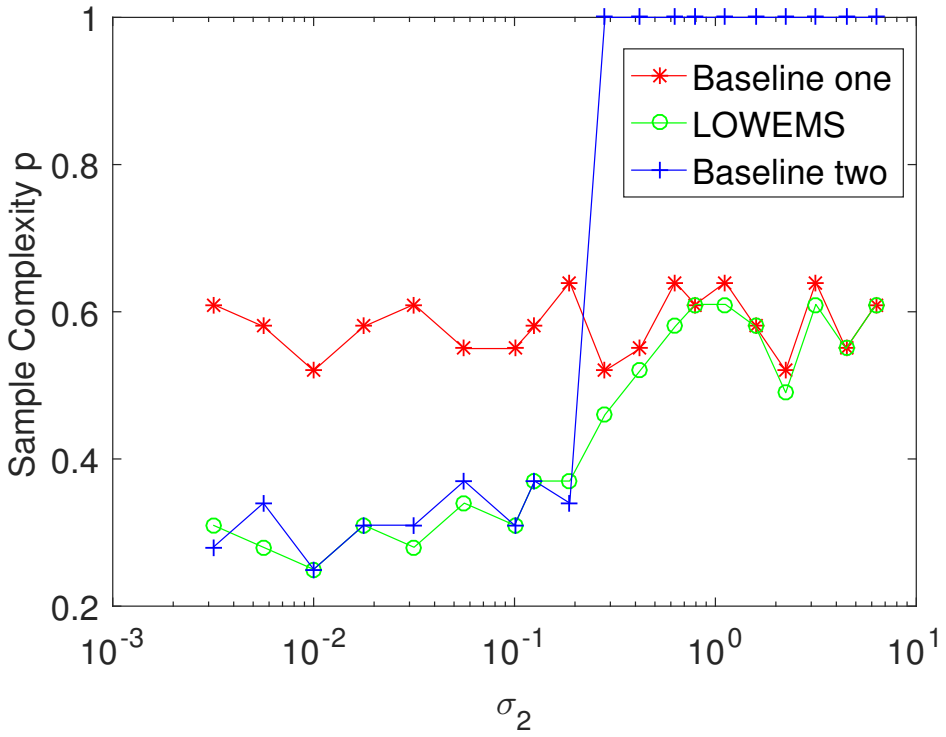

Figure 3.3: Sample complexity vs. perturbation noise ($\sigma_1 = 0.1$).

chronologically. As we can see from Figure 3.4, exploiting the dynamic constraint yields better prediction performance on this dataset.

3.2 Faster simultaneous recovery

3.2.1 Introduction

In this section we extend the approach of [46] by designing a two-stage estimator in the context of estimating a sequence of low-rank matrices simultaneously under a discrete random walk model.

3.2.2 Problem Formulation

Following the same setup in Section 2.2, our problem is to recover the sequence $\{X^t\}_{t=1}^d$ from $\{y^t\}_{t=1}^d$.

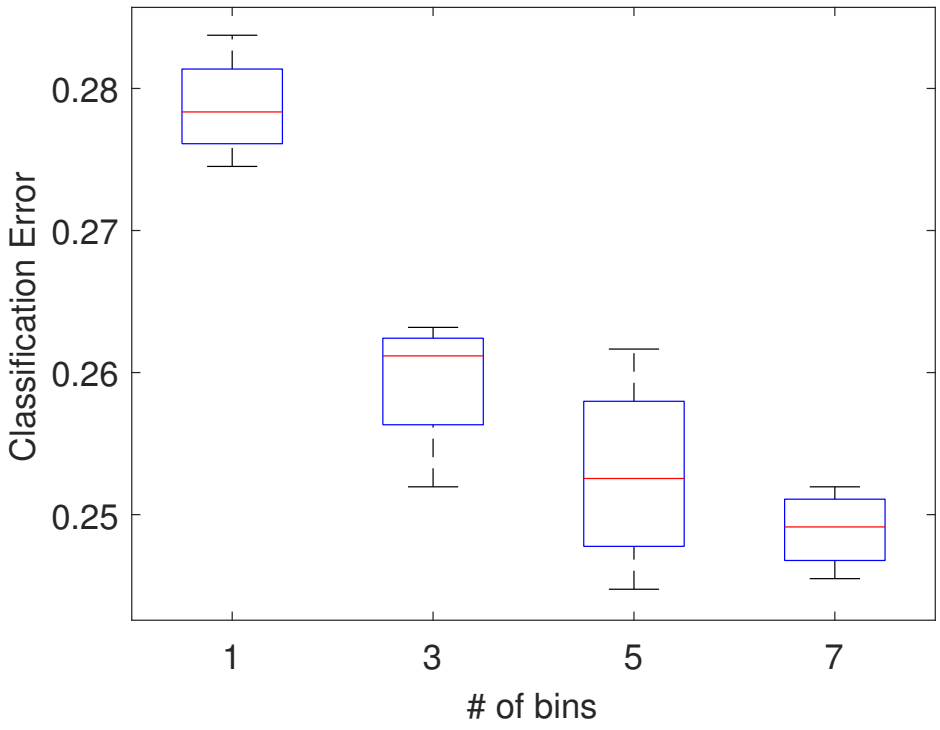

Figure 3.4: Experimental results on *ASSISTment* dataset

3.2.3 S-LOWEMS estimator

Maximum likelihood estimator

The first approach is to consider the recovery problem as a latent factor learning problem. A maximum likelihood estimator (MLE) is given by minimizing the following negative log-likelihood:

$$\begin{aligned}\mathcal{L}(U, V^1, \ldots, V^d) = \frac{1}{\sigma_1^2} \sum_{t=1}^{d} \left\| \mathcal{A}^t \left(U V^t \right) - y^t \right\|_2^2 \\ + \frac{1}{\sigma_2^2} \sum_{t=2}^{d} \left\| V^t - V^{t-1} \right\|_F^2 .\end{aligned} \tag{3.5}$$

The above cost function consists of two terms: the first term quantifies data fidelity and the second term quantifies the dynamic constraint on V. Although minimizing (3.5) is a nonconvex optimization problem, we can attempt to solve it via the alternating least squares

(ALS) algorithm over U and $\{V^t\}_{t=1}^d$. Note however that in this case the convergence of the ALS algorithm is not (known to be) guaranteed and the computational burden is quite heavy (especially when d is large, since we need to update all V^t's at each update).

A fast estimator based on weighted smoothing

In this section we use the idea of weighted smoothing from [46] to form a fast estimator of $\{X^t\}_{t=1}^d$. We first introduce the LOWEMS estimator proposed in [46], which is an algorithm that aims to produce an estimate of only X^d from $\{y^t\}_{t=1}^d$. The LOWEMS estimator consists of solving the following optimization program:

$$\widehat{X}^d = \underset{X \in \mathbb{C}(r)}{\arg\min} \frac{1}{2} \sum_{t=1}^{d} w_t \left\| \mathcal{A}^t(X) - y^t \right\|_2^2, \tag{3.6}$$

where $\mathbb{C}(r) = \{X \in \mathbb{R}^{n_1 \times n_2} : \text{rank}(X) \leq r\}$, and $\{w_t\}_{t=1}^d$ are non-negative weights with constraint $\sum_{t=1}^d w_t = 1$. If we define $\kappa := \sigma_2^2/\sigma_1^2$ and set $p_t = (d-t), 1 \leq t \leq d$, then one can calculate the optimal weights as [46]:

$$w_t^* = \frac{1}{\sum_{i=1}^d \frac{1}{1+p_i\kappa}} \frac{1}{1+p_t\kappa}, \quad 1 \leq t \leq d.$$

The parameter κ measures how strong the perturbation noise is compared to the observation noise.

Note that one can modify LOWEMS to recover X^s for any $s \in [d]$ by solving the following similar program:

$$\widehat{X}^s = \underset{X \in \mathbb{C}(r)}{\arg\min} \frac{1}{2} \sum_{t=1}^{d} w_t^s \left\| \mathcal{A}^t(X) - y^t \right\|_2^2, \tag{3.7}$$

where $\{w_t^s\}_{t=1}^d$ are a different set of weights to be used when estimating X^s. Following

similar arguments in [46], the optimal weights in this case are:

$$w_t^{s*} = \frac{1}{\sum_{i=1}^{d} \frac{1}{1+p_i^s \kappa}} \frac{1}{1+p_t^s \kappa}, \quad 1 \leq t \leq d, \tag{3.8}$$

where $p_t^s = |t-s|$.

A naïve extension of the LOWEMS method to recover X^s for all $s \in [d]$ is to perform program (3.7) independently for each $s \in [d]$. However this approach does not take into account the fact that for all $s \in [d]$, X^s should share the same U. This clearly leaves some room for potential improvement. Moreover, because the weights in (3.8) are selected specifically to minimize the recovery error for a particular X^s, the weights necessarily "downweight" previous/future observations. This can be helpful in obtaining a more accurate estimate of V^s, but this can actually be harmful in terms of our estimate of U (since it is essentially using only a small subset of the data in its estimate).

Inspired by this observation, we consider an alternative method which, although still quite simple, has the potential to improve on the naïve approach described above. Specifically, we conjecture that an equal weighting will yield an improved estimate of U (or more precisely, the column space of U) compared to the results of using (3.8) for any particular choice of s. Thus, we first estimate U from $\{y^t\}_{t=1}^d$, and we then follow this step by estimating $\{V^t\}_{t=1}^d$ by solving (3.7) using (3.8) while holding U fixed. This approach is summarized as follows:

Algorithm 1 S-LOWEMS: Simultaneously Locally Weighted Matrix Smoothing

1: Given d, κ, $\{y^t\}_{t=1}^d$ and $\{\mathcal{A}^t\}_{t=1}^d$
2: Solve (3.7) with equal weights to obtain $\widehat{U}$
3: For each $s \in [d]$, solve (3.7) via least-squares with $U = \widehat{U}$ and w_t^{s*} in (3.8) to obtain $\widehat{V}^s$
4: Output the estimate $\widehat{X}^s = \widehat{U}(\widehat{V}^s)^T$ for all $s \in [d]$

Remark 3.2.1. One can solve (3.7) in step 2 via alternating minimization (see e.g., [57]) or gradient descent (see e.g., [62]) based on matrix factorization.

Remark 3.2.2. Compared to MLE, Algorithm 1 is solving a bi-convex relaxation of the objective in (3.5).

Remark 3.2.3. Compared to the naïve extension of LOWEMS, the computational complexity of S-LOWEMS is actually quite small. Instead of increasing the computational complexity over a single LOWEMS by a factor of d, we need only perform a single LOWEMS and then the only additional computational overhead involves solving d least-squares problems. The same is true when comparing to MLE-ALS; we do not need to update all V^t's at each update, which saves both storage and computation.

Remark 3.2.4. We conjecture that for each $t \in [d]$, the recovery error of S-LOWEMS is smaller than that of a single LOWEMS estimator. However, we leave the proof of this conjecture for future work.

3.2.4 Simulations and Experienments

Synthetic simulations

In the following synthetic simulations we restrict our attention to matrix completion, although we expect similar results for other observation models. We use the relative recovery error (RRE) at time d, i.e., $\|\widehat{X}^d - X^d\|_F^2/\|X^d\|_F^2$, as our recovery accuracy metric (similar results are obtained when we look at the full sequence $\{X^t\}_{t=1}^d$). We set $n_1 = 100$, $n_2 = 50$, $d = 4$ and $r = 5$. We first generate entries of U and V^d uniformly from $[-0.5, 0.5]$ and $\{V^t\}_{t=d-1}^1$ according to (2.2). We orthonormalize U afterwards and generate y^t according to (2.1). For the purpose of illustration we consider two additional baselines besides LOWEMS and MLE-ALS: **baseline one** is the MLE assuming all V^t's have no dynamic constraints (hence σ_2 is infinity); **baseline two** is the MLE assuming all V^t's are the same (hence σ_2 is zero).

1). Recovery error. We set $\sigma_1 = 0.05$. In the first simulation, we vary the perturbation noise level σ_2 while keeping $m_0 = 4000$. For every σ_2 we perform 10 trials, and show the average RRE. As one can see from Figure 3.5, when σ_2 is small, all the three estimator LOWEMS, MLE-ALS and S-LOWEMS achieve almost the same RRE as baseline two. As σ_2 grows, the RRE of LOWEMS will increase due to the perturbation noise. However in

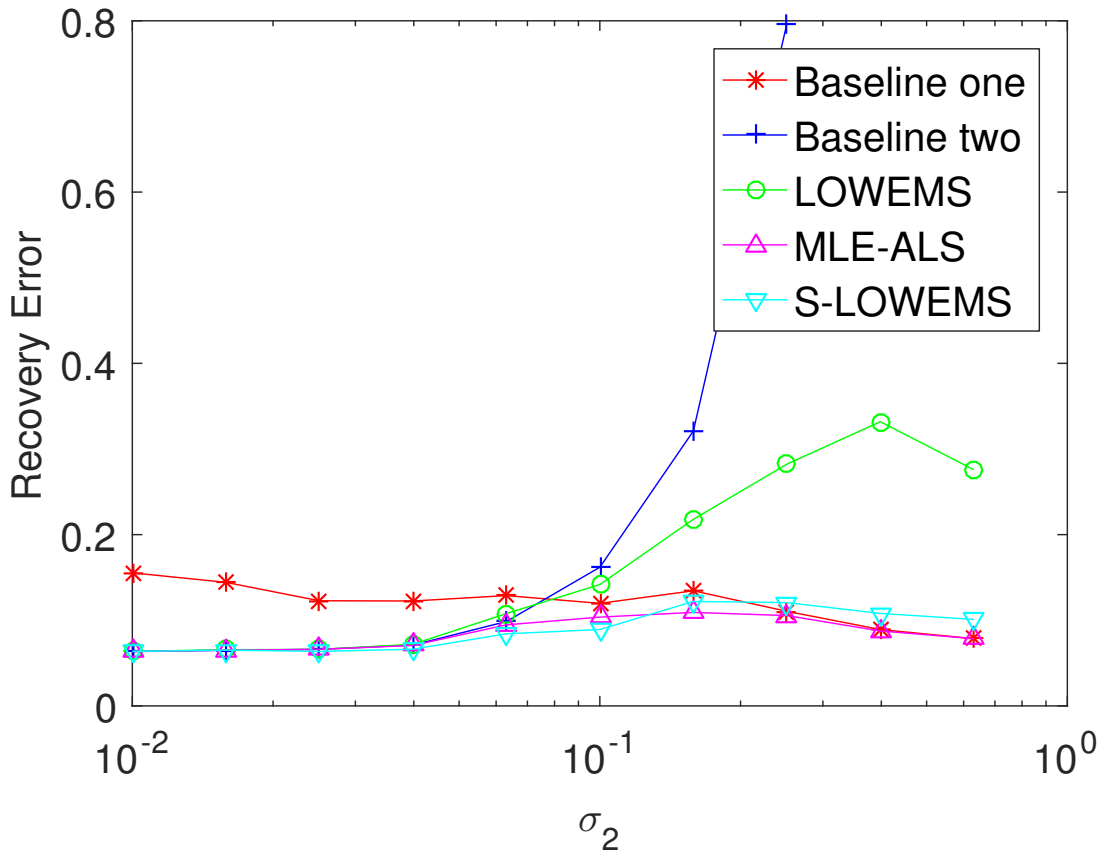

Figure 3.5: Recovery error under different levels of perturbation noise.

this case both S-LOWEMS and MLE-ALS achieve smaller RRE compared to LOWEMS. Notice that only when σ_2 is relatively large (compared to the matrix V itself, say 0.3) the RRE of S-LOWEMS is slightly larger than that of MLE-ALS. We also note that increasing perturbation noise (from 0.2 to 0.6) decreases the RRE of MLE-ALS. The reason is that the perturbation noise is large enough to help the recovery of U, and in turn reduce the RRE of recovering X^d (though we suspect this would be rare in practice).

In the second simulation, we vary the fraction of observed entries $p := m_0/(n_1 n_2)$ while keeping $\sigma_2 = 0.2$ (moderate). From Figure 3.6, we can see that S-LOWEMS almost achieves the best RRE (comparable to MLE-ALS) under various p.

2). Sample complexity. In this simulation we vary p to empirically find the minimum sample complexity required to guarantee successful recovery ($RRE \leq 0.06$). We compare the sample complexity of LOWEMS, MLE-ALS and S-LOWEMS under various σ_2 (σ_1

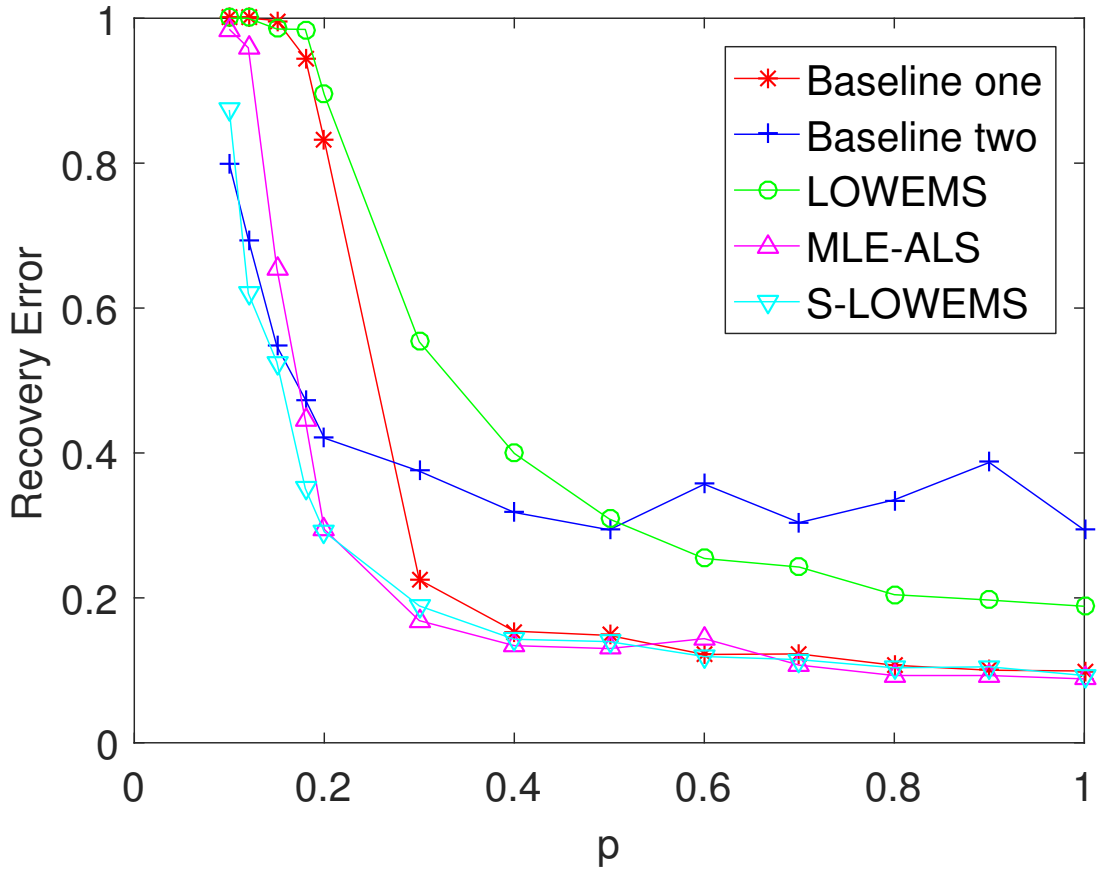

Figure 3.6: Recovery error under different percentages of missing entries.

is set as 0.02). For a fixed σ_2, the RRE is averaged over 10 trials. From Figure 3.7, we can see when the perturbation noise is small (less than 0.04), the sample complexities of LOWEMS, MLE-ALS and S-LOWEMS are almost the same as baseline two. When the perturbation noise increases, the RRE of the three estimators will increase due to the perturbation noise and hence the sample complexity increases. As we can see, in this case S-LOWEMS achieves a smaller sample complexity compared to LOWEMS and a bit larger than that of MLE-ALS (the price paid for not forming a MLE).

In general, our synthetic simulations demonstrate that the proposed S-LOWEMS achieves better performance (in terms of recovery error and sample complexity) than LOWEMS, and comparable performance as MLE-ALS with less computation and storage.

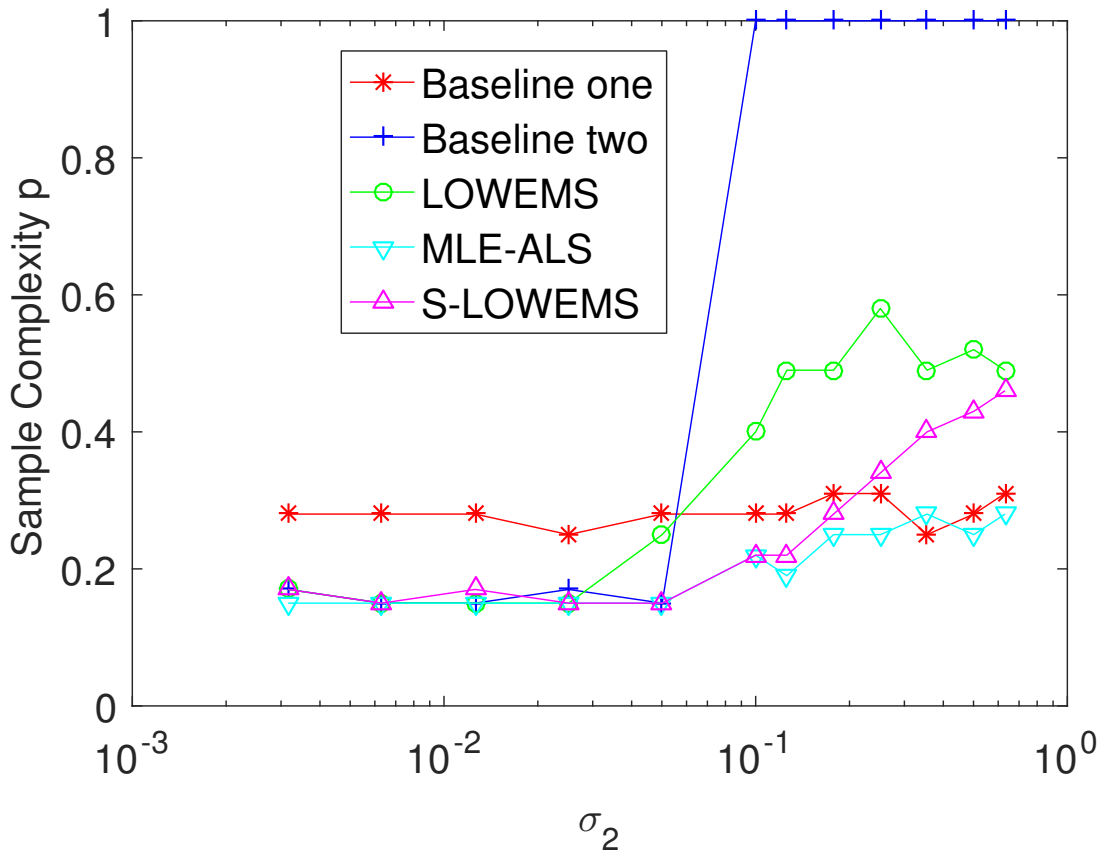

Figure 3.7: Sample complexity under different levels of perturbation noise.

Real world experiments

We next test LOWEMS, MLE-ALS, and S-LOWEMS in the context of a recommendation system using the (truncated) Netflix dataset. We eliminate those movies with few ratings and those users rating few movies, and generate a truncated dataset with 3199 users, 1042 movies, and 2462840 ratings. In this case the fraction of visible entries in the rating matrix is ≈ 0.74. All the ratings are distributed over a period of 2191 days.

For the sake of robustness, we additionally impose a Frobenius norm penalty on the factor matrices U and V. We keep the latest (in time) 10% of the ratings as a testing set. The remaining ratings are split into a validation set and a training set for the purpose of cross validation. We divide the remaining ratings into $d \in \{1, 3, 6, 8\}$ bins respectively according to their timestamps so that each bin contains the same number of ratings (see Figure 3.8). We

use 5-fold cross validation, and we keep 20% of the ratings from the d^{th} bin as a validation set. The number of latent factors r is set to 10. The Frobenius norm regularization parameter γ is set to 1. We also note that in practice one likely has no prior information on σ_1, σ_2 and hence κ. However, we use model selection techniques like cross validation to select the best κ incorporating the unknown prior information on measurement/perturbation noise. We use root mean squared error (RMSE) to measure prediction accuracy. Since alternating minimization uses a random initialization, we generate 10 test RMSE's. Figure 3.9 shows that all the three temporal estimators LOWEMS, MLE-ALS and S-LOWEMS improve the testing RMSE with appropriate κ compared to the static baseline (when $d = 1$). In addition, the testing RMSE of S-LOWEMS is lower than that of LOWEMS and MLE-ALS in general. Our results show that exploiting the fact that the user factor matrix V is changing yields improved prediction performance.

3.3 Conclusions

In this chapter, we extended LOWEMS to two additional settings. First, to accommodate applications where a linear observation model is not appropriate and we may only have access to binary observations, we proposed the one-bit LOWEMS algorithm and demonstrated its performance by synthetic simulations and by real-world experiments in the context of personalized learning. Second, in order to recover quickly and simultaneously a series of low-rank matrices, we proposed the S-LOWEMS estimator, and we analyzed its recovery performance by synthetic simulations and tested it on the truncated Netflix dataset. Our results show that, compared to the original LOWEMS estimator, the proposed S-LOWEMS estimator not only recovers a series of low-rank matrices simultaneously with a small computational overhead but also improves the recovery accuracy and sample complexity. Furthermore, the proposed S-LOWEMS estimator achieves almost the same statistical efficiency as the MLE (especially when the perturbation noise is small or moderate) and consumes significantly less storage and computational resources. However, our model has

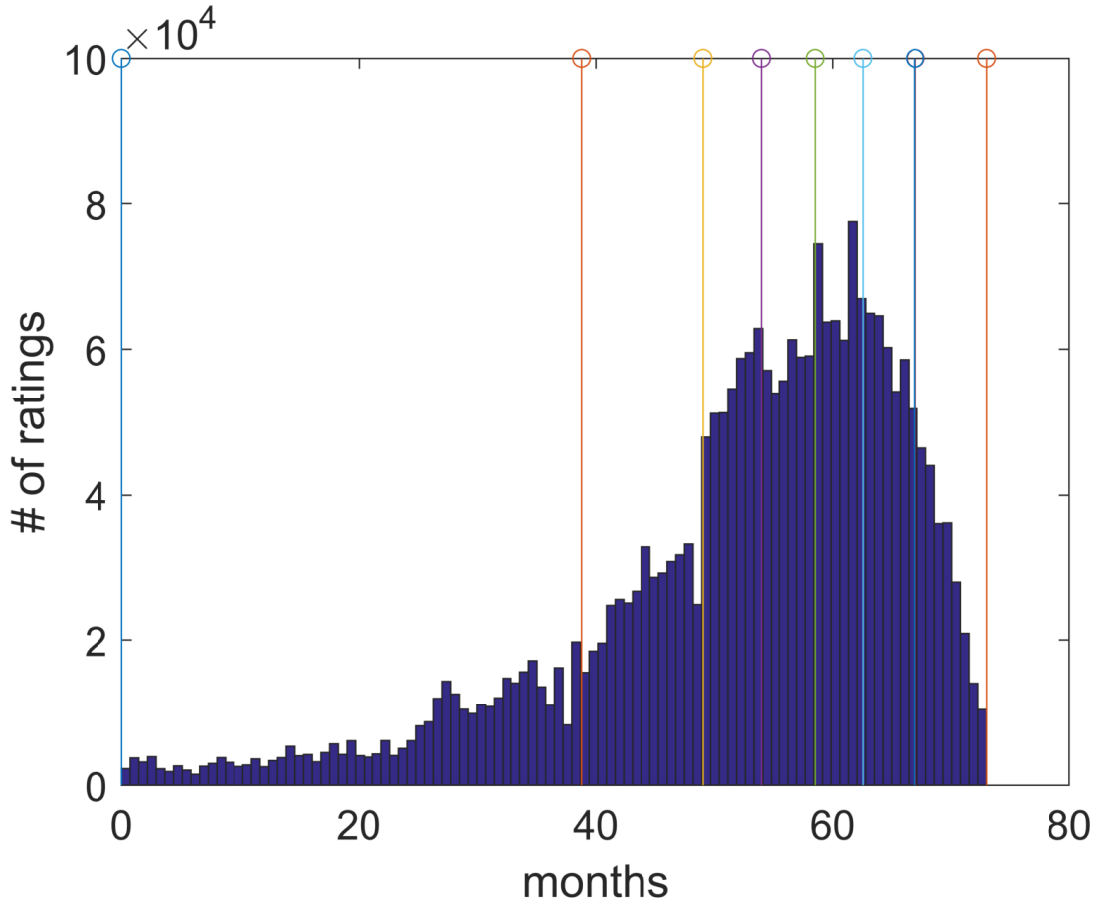

Figure 3.8: Ratings divided into 7 bins (6 for training and 1 for testing) on the truncated Netflix dataset.

several limitations. For example, we assume a random walk model on one of the factor matrices. Some possible future extensions of this work include more sophisticated dynamics models and a theoretical analysis to obtain provable recovery guarantees.

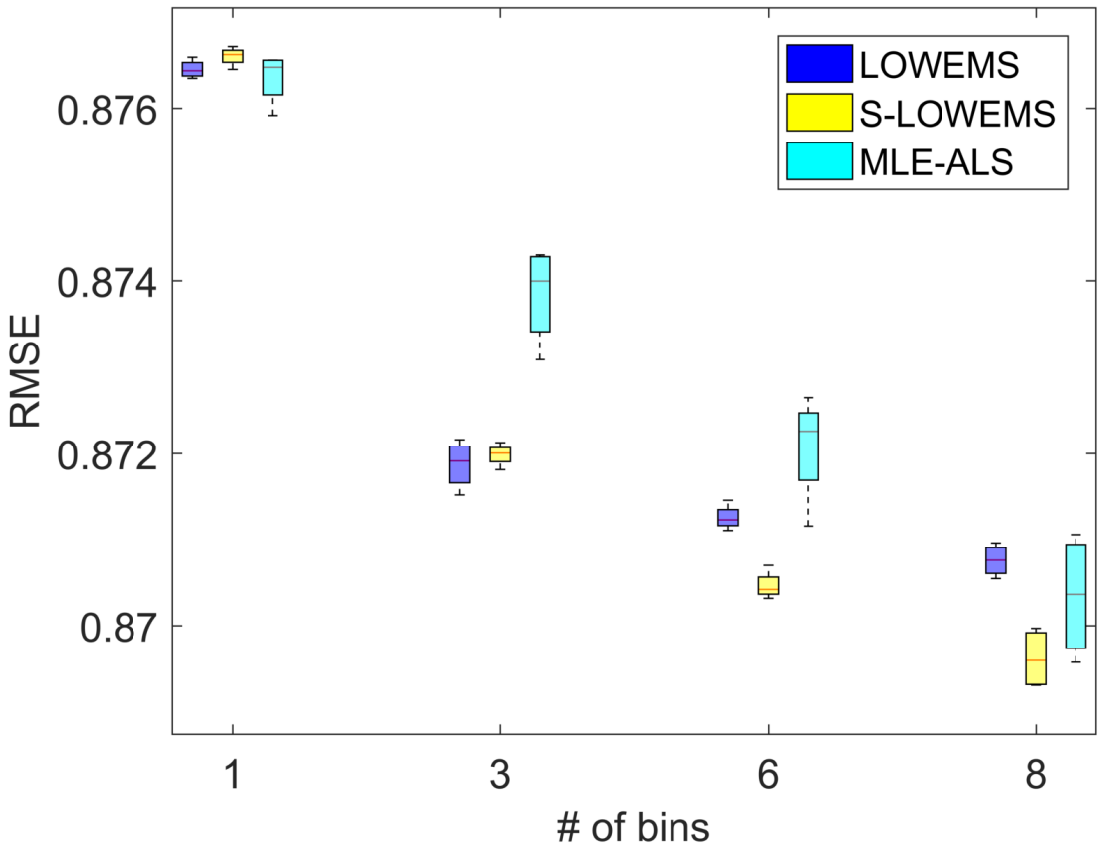

Figure 3.9: Experimental results on the truncated Netflix dataset: prediction RMSE vs. number of time bins.

CHAPTER 4

LOW-RANK MATRIX RECOVERY FOR MEASUREMENT INDUCED DYNAMICS: DYNAMIC KNOWLEDGE EMBEDDING AND TRACING

Although the random walk dynamics model can be effective in practice, it is not always the best way to describe the dynamics of a low-rank matrix recovery problem. For example, consider the dynamics of student learning. It is reasonable to assume that the evolution of students' knowledge is affected at least in part by the measurement process (answering questions), rather than being purely random. For example, suppose that when a student answers some questions, he learns some relevant knowledge by reviewing textbooks or checking the reference solutions. More precisely, the interaction between students and questions changes the students' knowledge in a manner related to the concepts that the questions contain. This phenomenon is also observed in other situations, such as recommendation systems and survey response data. In this chapter, we investigate the dynamic low-rank matrix recovery problem under measurement induced dynamics from a practical perspective in the context of knowledge tracing.

4.1 Introduction

A central component in many computer-based learning systems, and in any kind of *intelligent tutoring system* (ITS), is a method for estimating and tracking a student's knowledge or proficiency based on the student's previous interactions with the system. For example, a student may interact with many different course materials (homework exercises, quiz/exam questions, textbooks and other course materials, etc.) over a potentially long period of time. As a result of these interactions (and due to the passage of time) the student's knowledge and proficiency will dynamically evolve over time [75, 76, 77, 78]. Tracking the state of a student's knowledge as it evolves can provide deeper understanding how the student is

learning and which interactions (questions, textbooks, etc.) are most helpful, ultimately enabling the creation of a personalized learning environment tailored to provide an improved learning experience for the student.

Estimating student knowledge or proficiency from a sequence of student interactions poses two fundamental challenges. First, student proficiency evolves over time as the student interacts with the system. For example, the student might turn to textbooks or seek the teacher's help in response to getting a particular question wrong, and then may be able to answer a similar question correctly afterwards. Alternatively, the student may gradually lose proficiency in some areas if long periods of time pass without using this knowledge (e.g., over summer breaks or long vacations). Thus, we cannot treat this as a static problem of estimating a student's knowledge, but must think of this as a dynamic tracking problem. A second and more subtle challenge is posed by the fact that the manner in which student proficiency evolves may be strongly influenced by the nature of the interactions. For example, when a student is posed a question that requires knowledge of a particular concept, we not only learn something regarding the student's proficiency, but the student may also also learn something from the question. In this way, the interactions both provide information to help us track the student's knowledge while simultaneously inducing changes in the state that we wish to track.

In this chapter we propose a framework for tracing student knowledge using only a sequence of student responses to questions (for an ensemble of many students). The framework consists of two core components: a (static) embedding network that learns fixed latent representations of questions from student-question interactions and a recurrent neural network (RNN) that dynamically tracks the hidden state corresponding to each student's knowledge over time from the student's sequence of interactions. Our main contributions are:

- A new knowledge tracing framework which exploits both the advantages of latent question embedding from response data and an RNN to track student knowledge;

- A framework that can track student knowledge without using the question-level concept/skill tags that other knowledge tracing models (e.g., DKT [76] and its variants) require, avoiding labor-intensive manual tagging;
- A flexible framework that can also accommodate a variety of sequential modeling techniques (e.g., memory networks [79]) and can incorporate tag information and other features when available.

4.2 Related work

4.2.1 Educational data mining

Extracting useful information from the kind of educational data we consider was first studied within the *intelligent tutoring* community. Since the seminal work of [75], there has been a variety of efforts aimed towards understanding the cognitive processes that are most relevant in the context of an ITS, most of which aim to estimate students' proficiency based on their past interactions with the system with the aim of predict their performance on the new exercises/tests or customizing their learning materials.

Static models. Item Response Theory (IRT) is a standard framework for modeling student responses to questions dating back to the 1950s [80]. Perhaps the most common IRT model is the Rasch model [81]. This is a simple two-parameter model in which each student is modelled as having a particular skill level and each question has a particular difficulty, which is then paired with a logistic link function to provide predictions of the probability a student will answer a question correctly. There are natural mutlidimensional extensions of this and similar IRT models, which can be viewed as special cases of standard matrix factorization models ([82]) or more general factorization machine model [83]).

Perhaps the most effective of such static models is the Knowledge Tracing Machine (KTM) model of [84]. The core idea for the KTM model is to model the probability of a correct response via a (sparse) weighted combination of features as well as the interactions

amongst those features in a way that is mediated by a learned embedding. This model contains several models as specials cases, including many standard IRT models, the additive factor model (AFM, [77]), performance factor analysis (PFA, [78]), and matrix factorization (MF, [82]). It is worth noting that while the KTM framework does not explicitly incorporate any notion of dynamics, it is possible to implicitly do this by including past student-question interactions (correct or incorrect responses) as additional features. In fact, both AFM and PFA incorporate this kind of information.

Sequential models. Most of the models described above, at their core, involve estimating a fixed student-question embedding which is then used to predict future responses. However, we fully expect the state of a student's knowledge to change over time. To capture such dynamics, a natural approach is to more explicitly incorporate the interaction history in our model. One of the most popular models is Bayesian Knowledge Tracing (BKT), which employs a hidden Markov model ([75]) to model the process of mastering a particular skill. However, the BKT approach has some significant drawbacks. Most significantly, it models only a single skill or concept at a time. In practice, any particular question may be associated with a complex combination of different skills (to varying degrees). The BKT framework is limited in its ability to accommodate such settings. To overcome this shortcoming, a number of alternative approaches have recently been proposed.

The most relevant attempt in this direction is the Deep Knowledge Tracing (DKT) framework [76]. The DKT approach was inspired by recent progress in RNNs and deep RNN architectures. RNNs are a family of neural networks tailored for sequential prediction problems. They are recursive in the sense that the encoded hidden state evolves based both on the input as well as the network's previous states [85]. In recent years deep RNN architectures have been shown to outperform many classical models in many application areas, including natural language processing and session-based recommendation system. DKT is the first model to use RNNs to track student knowledge. DKT uses a one-hot

encoding of skill/concept tags and associated responses as input and trains the RNN to predict the future student response.

However empirical experiments in [86, 87, 88] show that DKT does not appear to result in substantial improvement over many simpler models from classical IRT whose parameters and inferred states are psychologically meaningful. It is worth noting that the IRT variants considered in [86, 87, 88] use problem IDs as identifiers instead of skill IDs for DKT. Since multiple problem IDs can be tagged with the same skill IDs, we generally find that skill IDs repeat much more frequently than problem IDs. Thus, a comparison using skill IDs would likely be more favorable to a recurrent/sequential model like DKT. Of course, in considering only skill IDs we lose the ability to learn/exploit question-level information such as question difficulty. Moreover, producing skill IDs for each question requires substantial human effort and is often not feasible in practice. Furthermore all the experiments in [86, 87, 88] consider the 'New Student' evaluation protocol, which keep a portion of the students as training sets and test on new students. Such an evaluation scenario may not be particularly meaningful in a real-world ITS and does not favor penalization models such as IRT, though online evaluation in [87, 88] mitigates such bias. Thus, the comparison study in [86, 87, 88] is not entirely satisfying and leaves open many questions regarding the potential benefits (or lack thereof) of deep RNNs for knowledge tracing.

Hybrid models. There are also several attempts to combine static models and sequential models to exploit advantages from both approaches, such as the FAST model in [89] and the LFKT model in [90]. Although the two models are described in different terms, they are in fact equivalent, with the main difference being their training method. In [91], these two approaches are compared and the experimental results show that these two hybrid models do not outperform a simple IRT model. The authors conjecture that the lack of improvement is due to a confounding between item identity and the question position in a (nearly deterministic) sequence of questions. In contrast to these more pessimistic results, in

this chapter we propose a hybrid model and show that it can harness the advantages from both static and sequential models in a way that outperforms both.

4.2.2 Session-based recommendation systems

A closely related application to knowledge tracing is that of predicting a user's preference for various items (movies, music, books, etc.) in a recommendation system. Among various recommendation systems, session based recommendation is the most closely related to knowledge tracing. For example, a session-based recommendation model, GRU4Rec, is proposed in [92] that has a similar architecture as DKT. However, GRU4Rec does not consider user identifications as inputs. An alternative approach – the Recurrent Recommender network (RRN)[93] – is capable of both modelling the seasonal evolution of items and tracking the user preferences over time. RRN uses a matrix factorization to model the stationary component of the user and item embeddings, and then two Long Short-Term Networks (LSTMs) to track the dynamic component of these embeddings.

Though similar, there are some notable differences between product recommendation and knowledge tracing. First, user preferences tend to change much more slowly compared to student knowledge. Second, student interactions with questions have a significant impact on student knowledge, while in contrast interactions with an item (watching a movie, buying a product, etc.) typically has a mild impact at most on user preferences. Third, in a recommendation context, user responses may contain important implicit feedback [94]. For example, we can conclude that a user will watch a movie or buy a product because he/she likes it, even if the user does not give explicit feedback. However, students typically have limited freedom to choose which questions to answer. Moreover, these questions are typically sequenced in a way that is very far from random (and much less random than the kinds of activities observed by typical recommendation system). These differences have important algorithmic implications.

4.3 The *DynEmb* framework

4.3.1 System architecture

In this section we describe a novel framework for tracking student knowledge, dubbed *DynEmb*, that learns a *static* question embedding but tracks the knowledge state by exploiting *sequential* models of the temporal dynamics of student-question interactions. We will represent our training data as a sequence of interactions of the form $\mathcal{R}_t = (s_t, q_t, r_t, o_t)$. Each interaction $\mathcal{R}_t$ involves a student s_t and a question q_t. The response to the question is denoted r_t, which is most commonly a correct/incorrect binary outcome or occasionally a numerical score. We assume there are M questions and N students. In this chapter we focus mainly on the binary case, but the underlying framework can easily extend to the more general setting. Finally, we let o_t denote other information about the interaction that may be relevant, including – but not limited to – time stamps, questions tags, platform (e.g., paper, computer, mobile, etc.), and question text descriptions.

The goal of *DynEmb* is to predict student responses to future questions given a historical sequence of interactions $\{\mathcal{R}_i\}_{i=1}^{n}$. Specifically, given a new student-question pair (s_t, q_t) and any additional information o_t if available, our goal is to predict r_t. *DynEmb* has two main components, each of which are trained independently (see Figure 4.1). The first component *QuestionEmb* generates a $d-$dimensional *question embedding* $W_{q_t} \in \mathbb{R}^d$ from $\{\mathcal{R}_i\}_{i=1}^{n}$ using standard matrix factorization techniques described in more detail below. The second component *StudentDyn* learns to track each student's knowledge state using a sequential model that takes the student's past sequence of question embeddings $\{W_{q_i}\}_{i=1}^{t-1}$ and responses $\{r_i\}_{i=1}^{t-1}$ as inputs and produces a dynamic student embedding $Z_{s_t}(t) \in \mathbb{R}^d$. The sequential model could be a "vanilla" RNN, a long short-term memory (LSTM) network, a gated recurrent unit (GRU), a memory network with attention, or others. In this work we use an LSTM in the *StudentDyn* component by default. After obtaining the (static) question embedding W_{q_t} and the (dynamic) student embedding $Z_{s_t}(t)$, the predicted probability of a

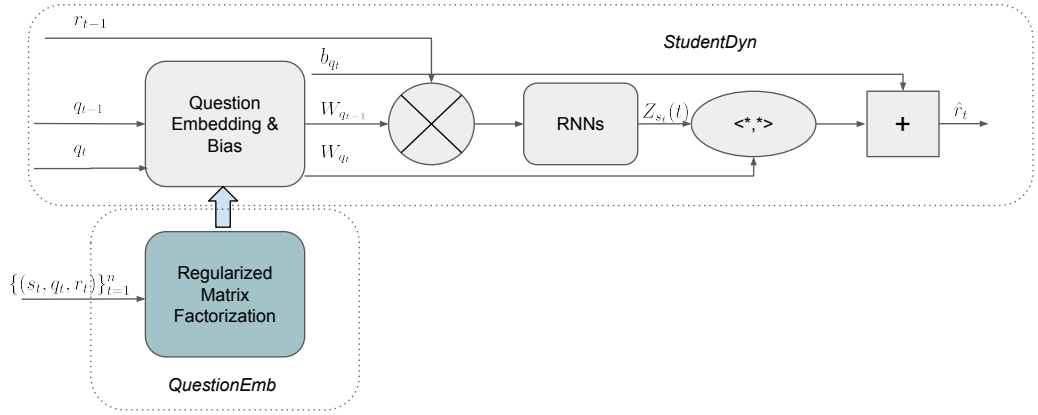

Figure 4.1: Architecture for *DynEmb*. First we train *QuestionEmb* to obtain question embedding W and bias b. Then we train the RNNs using past item embedding $W_{q_{t-1}}$ and response r_{t-1} as inputs to track student knowledge.

correct response is computed via

$$\hat{r}_t = \phi\left(\langle W_{q_t}, Z_{s_t}(t)\rangle + b_{q_t}\right), \tag{4.1}$$

where b_{q_t} is a scalar that represents a bias learned for each question and ϕ is a sigmoid activation function. We describe these components in further detail below.

QuestionEmb. The *QuestionEmb* component uses an ℓ_2-regularized biased matrix factorization model to learn a static latent embedding for the questions. More specifically, in this component we learn both a question embedding W and a student embedding Z, where $W \in \mathbb{R}^{N\times d}$ is a matrix whose columns correspond to a question's embedding vector W_q and $Z \in \mathbb{R}^{M\times d}$ is a matrix whose columns correspond to a student's embedding vector Z_s. These are learned via the following optimization problem:

$$\begin{aligned}\underset{W,Z,b,c}{\arg\min} \sum_{t=1}^{n} \mathcal{L}\left(r_t, \phi\left(\langle W_{q_t}, Z_{s_t}\rangle + b_{q_t} + c_{s_t}\right)\right) \\ + \lambda\left(\|W\|_F^2 + \|Z\|_F^2\right),\end{aligned} \tag{4.2}$$

where b and c are vectors of question and student "biases" respectively, λ is the regularization parameter, and $\mathcal{L}(y, x) = -(y \log(x) + (1 - y) \log(1 - x))$ is the log loss function. This is inspired by the observations in [69] that if the question embedding W is static, then one can still use conventional matrix factorization to recover W, even though the other factors Z may actually be changing over time. Finally, we note that while (4.2) is a non-convex optimization problem, simple optimization algorithms exist that provably converge to a global minimum [57, 95].

StudentDyn. The *StudentDyn* component uses an RNN to sequentially generate a student embedding after each interaction. For the case of a binary response, r_{t-1}, the input to the recurrent neural network is the Kronecker product of the question embedding learned by the *QuestionEmb* component ($W_{q_{t-1}}$) and the vector $[r_{t-1}, 1 - r_{t-1}]^T$. At time step t, an interaction between student s_t and question q_t is predicted via the model in (4.1), and the RNN is trained to predict r_t. The dynamic student embedding $Z_{s_t}(t)$ is the internal hidden state of the RNN, which is then combined with W_{q_t} via (4.1) to obtain our final prediction.

4.3.2 Model training

To train *DynEmb*, we adopt a two-phase pretraining strategy. We first train the question embedding in the *QuestionEmb* component. We then feed the learned question embedding to the *StudentDyn* component to train the sequential model. Note that we keep the question embedding W and the biases b fixed when training the *StudentDyn* component. This embedding pretraining strategy not only speeds up the training process, but also produces better prediction performance compared to end-to-end training (see Section 4.4.4 for an experimental justification). Similar pretraining strategies are widely used in learning complex models (e.g., for machine translation [96] and sentiment analysis [97]).

Compared to DKT [76], DKVMN [79], and other sequential knowledge tracing models, the explicit question embedding learned directly from interactions based on matrix factoriza-

tion seems to be more robust. In fact, in our experiments we have observed that if we replace the (frequently repeating) concept/skill tags in DKT and DKVMN with the (much less frequently repeating) question identifiers, then both DKT and DKVMN will have significant performance degradation and require intensive computational resources to train. However, our model can track student knowledge using the pretrained question embedding instead of concept/skill tags. This allows our approach to exploit question difficulty information and scales well, especially when concept/skill tags are not available.

4.3.3 Integrating skill tag information

If manually-labeled skill tag information is available for each question, then it is convenient and beneficial to incorporate this information into the *DynEmb* framework. However the question latent space learned via the matrix factorization might be different from the latent space constructed by manual labeling. One simple method to exploit both approaches consists of concatenating the two latent question embeddings to form a new latent question embedding. The skill tags can be one-hot encoded. To further exploit the hierarchical relationship between questions and skill tags, we initialize a question's embedding by the one-hot encoding of its corresponding skill tag, and put an additional ℓ_1 regularization on the objective in (4.2) to promote sparsity (see (4.3)):

$$\begin{aligned} \underset{W,Z,b,c}{\arg\min} \sum_{t\in[n]} & \mathcal{L}\left(r_t, \phi\left(\langle W_{q_t}, Z_{s_t}\rangle + c_{q_t} + b_{s_t}\right)\right) \\ & + \lambda\left(\|W\|_F^2 + \|Z\|_F^2\right) + \mu\|W\|_{\ell_1}. \end{aligned} \tag{4.3}$$

This aligns the latent space of question embedding with the latent space formed by the skill tags.

To control the dimensionality of the latent space, the concatenated embedding is followed by a fully connected (FC) layer with ReLU activation:

$$W_q' = \text{FC}(\text{Concat}(W_q, T_c)). \tag{4.4}$$

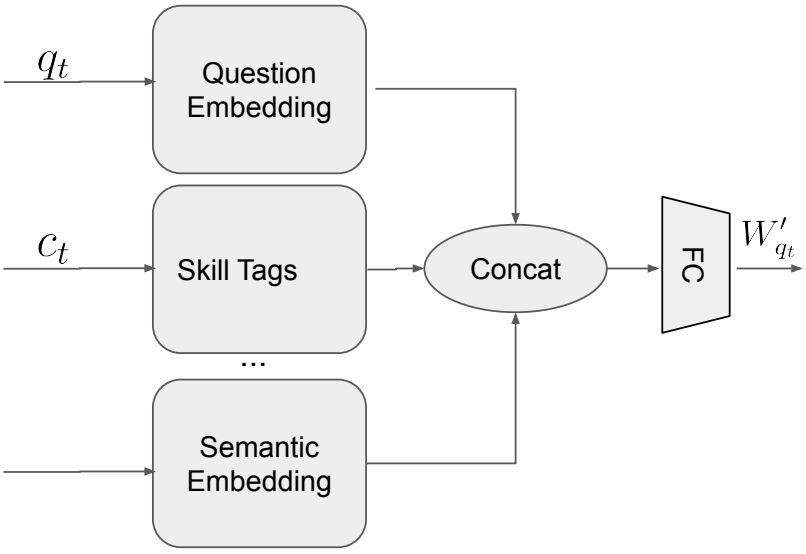

Figure 4.2: Multiple input fields. The concatenation layer takes multiple inputs and the FC layer fuses them to a form a single embedding.

This kind of integration scheme can be found in [98] and also enables easy incorporation of additional embeddings/fields, e.g., semantic embedding from question text.

Finally, the *StudentDyn* component uses an RNN to sequentially generate a student embedding after each interaction using this modified question embedding just as before. See Figure 4.2 for additional details.

4.4 Experiments

In this section, we experimentally validate the effectiveness of the proposed *DynEmb* model on two tasks: prediction of response correctness for existing students and prediction of response correctness for new students. By conducting experiments on several data sets each and comparing with the relevant baselines, we show that:

1. *DynEmb* outperforms DKT by up to 5.43% and 3.74% in predicting the next response in the 'New User' and 'Most Recent' evaluation settings respectively (see definition in Section 4.4.1);

2. The performance of *DynEmb* is stable with respect to the dimensionality of the item

embedding;

3. The proposed embedding pretraining strategy is a key component of the success of the *DynEmb* approach.

4.4.1 Experimental setting

We consider the following baselines:

- Algorithms that compute a static embedding: in this category, we compared with BMF [82], which is a special case of KTM [84]. We compare to both offline and online BMF.
- Knowledge tracing based on RNNs: we compare with the state-of-the-art DKT algorithm [76].

Evaluation metrics. We report the Area Under the ROC Curve (AUC) for comparing the predicted probabilities of correctness for each response. AUC is threshold agnostic, and is widely used in the knowledge tracing literature.

Evaluation methods. We use two evaluation methods. The first is online response prediction for new users [76, 87]. In this setting, students are first split into training and testing populations. Each model is first trained on the training population. Then for each time $t > 1$ in each testing student's history, we train the student-level parameters in the model on a new student, including both the training population and the first $t - 1$ interactions of the student history, computing the probability that the t^{th} response is correct. In practice, we find that re-training and testing after each response is not computationally feasible for large datasets, in which case we perform online response prediction in batches. We denote this evaluation method the 'New User' setting. Our second method is to consider online response prediction for the the most recent interactions as in [87]. The procedure here, denoted the

Table 4.1: Overview of data sets.

Data set	Number of				Ratio of correctness	Description
	Skills	Problems	Students	Responses		
ASSISTments	101	13111	4003	214424	0.658	2009
	265	47124	28998	2623624	0.699	2012
Cognitive Tutor	90	210710	574	809693	0.767	Algebra I 2005
	488	580531	1338	2270384	0.772	Algebra I 2006
	494	207856	1146	3679188	0.888	Bridge to Algebra 2006

'Most Recent' setting, is the same as in the 'New User' setting except that we consider only the most recent interactions for our testing population as the testing data set.

4.4.2 Experiment 1: Future response prediction

In this experiment, the task is to predict students' response. The prediction task is: given all interactions up to time t, given the student s and question q involved in the interaction at time t, what is student s's response (correct/incorrect) to question q?

We use the following data sets to evaluate performance on this task:

- **ASSISTments.** This data set was gathered from ASSISTments's skill builder problem sets, where students learn by working on similar questions until they can respond correctly n (usually 3) times in a row [74]. We use two one the provided data sets, "ASSISTment09" and "ASSISTment12." Note that the authors updated "ASSISTment09" in 2017 (first found in [86]).

- **Cognitive Tutor.** In the 2010 KDD Cup Challenge, the PSLC DataShop released several data sets from Carnegie Learning's Cognitive Tutor in (Pre-)Algebra from the years 2005-2009 [99]. We use three of the "Development" data sets, "Algebra I 2005-2006," "Algebra I 2006-2007," and "Bridge to Algebra I 2006-2007."

Preprocessing of data sets. As noted in [87], there are multiple records duplicating a single interaction (represented by a unique *order_id* value) in "ASSISTment09." These duplicate rows arise when a single interaction is aligned with multiple skills. This provides

Table 4.2: Future response prediction experiment: Table comparing the performance of *DynEmb* (concatenating question and skill embedding) with baselines, in terms of AUC. *DynEmb* outperforms the best baseline by up to 5.43%. We also list the performance of *DynEmb* only with question embedding

Evaluation method	Model	BMF		DKT	DynEmb		Improvement
		offline	online		Question	Concat	
New User	ASSISTment09	0.67	0.686	0.727	0.725	0.739	1.65%
	ASSISTment12	0.694	0.717	0.709	0.722	0.736	2.65%
	Algebra I 2005	0.761	0.763	0.773	0.803	0.815	5.43%
	Algebra I 2006	0.761	0.786	0.808	0.805	0.821	1.61%
	Bridge to Algebra 2006	0.838	0.844	0.856	0.868	0.873	1.99%
Most Recent	ASSISTment09	0.706	0.727	0.661	0.738	0.727	0.00%
	ASSISTment12	0.67	0.696	0.71	0.692	0.714	0.56%
	Algebra I 2005	0.744	0.763	0.779	0.791	0.808	3.72%
	Algebra I 2006	0.761	0.782	0.801	0.813	0.822	2.62%
	Bridge to Algebra 2006	0.831	0.839	0.847	0.859	0.865	2.13%

DKT models access to the ground truth when making their predictions, which can artificially boost prediction results by a significant amount. We adopt two strategies to clean the data. The first is to discard rows duplicating a single interaction (as in [87]); the second is to combine these duplicating rows into a single row with a new skill tag as suggested by [86]. In this chapter we removed duplicate and multiple-skill repeated records in all data sets to ensure fairness for the purpose of comparison. We also removed "not original" records as suggested by [86]. We do similar cleaning operation on the other data set "ASSISTment12."

For the Cognitive Tutor data sets, we form problem identifiers from the concatenation of the "Problem Name" and "Step Name" fields.

Implementation details. The dimensionality of the input to the RNNs in *DynEmb* is fixed at 100. The ℓ_2 regularization parameter in the *QuestionEmb* component is chosen using cross-validation based on standard BMF. The hyper-parameters in the *StudentDyn* component are the same as DKT and chosen by cross-validation.

Results. Table 4.2 compares the results of *DynEmb* with the baseline. We observe that *DynEmb* significantly outperforms the best baseline in all datasets in terms of AUC on the

three datasets up to 5.43%.

4.4.3 Experiment 2: Robustness to embedding dimensionality

In this section, we study the effect of the dynamic embedding dimensionality on the tracking performance. In this study we use the "ASSISTment09" and Cognitive Tutor "Algebra I 2005" ("CT05" for short) datasets, which have the smallest number of interactions from the two tutoring systems respectively. The effect on other datasets is similar and omitted due to space constraints. We will test on the response prediction task. As we can see from Figure 4.3, the performance by AUC of *DynEmb* is quite stable over a wide range of embedding dimensionalities. This robustness is an additional attractive feature of our approach.

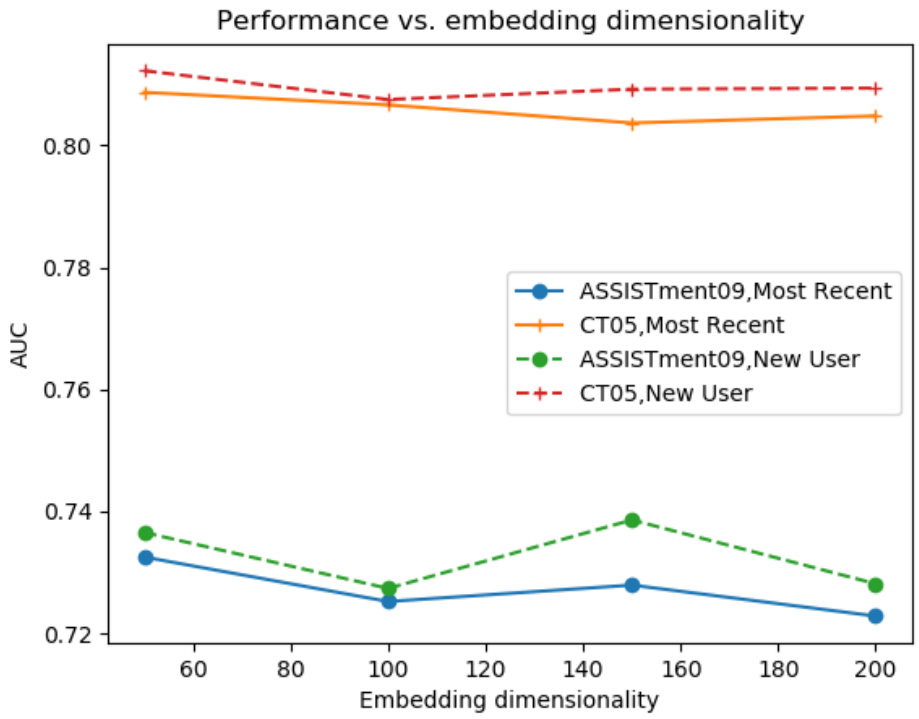

Figure 4.3: Performance versus embedding dimensionality.

4.4.4 Experiment 3: Embedding pretraining vs. end-to-end training

In this section we demonstrate why *DynEmb* uses pretraining for the question embedding. The dataset used in this section is "ASSISTment09." We use the "Most Recent" evaluation method. In Figure 4.4, we can see that end-to-end (E2E for short) training (with/without pretraining the question embedding) will cause over-fitting, while the learning curve of proposed pretraining strategy does not suffer from over-fitting or under-fitting. Of course, another advantage of pretraining is its improved computational efficiency. The combination of these two factors provides powerful evidence for choosing pretraining over an end-to-end training strategy in this framework.

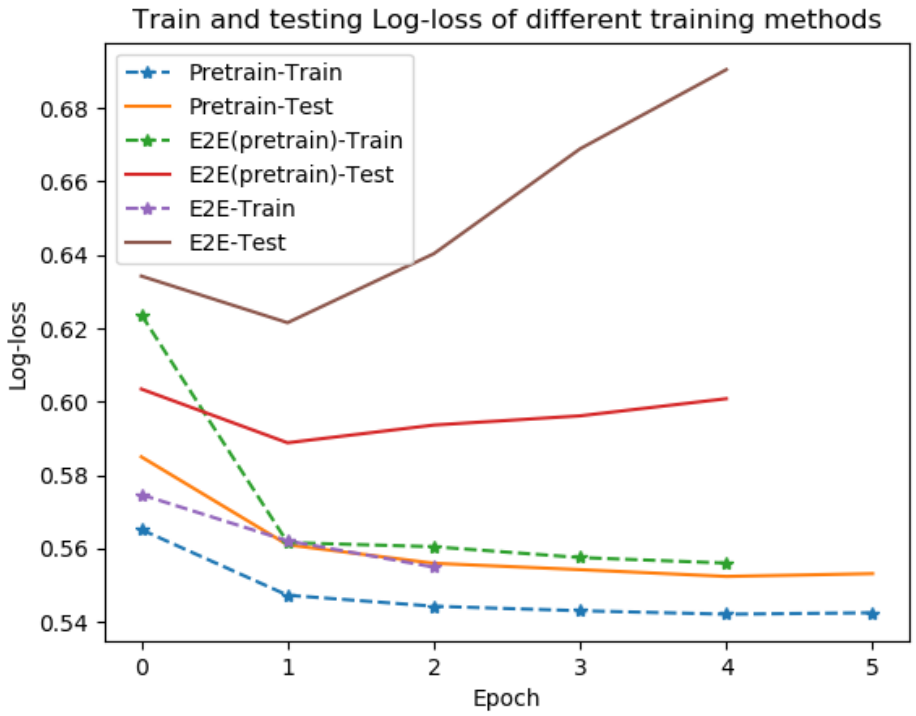

Figure 4.4: Training and testing log-loss of different training methods.

4.4.5 Experiment 4: Visualizing question embedding

Though the latent space of question embedding learned via matrix factorization is not explicitly aligned with the latent space formed by the manually-labeled skill tags that

were provided, the proposed question embedding initialization and sparsity promotion is remarkably effective at aligning the question embedding space with the manually constructed skill embedding space. This provides additional semantic meaning for the learned question embedding, which improves model interpretability. Figure 4.5 shows clear clustering of question embedding with respect to the associated skills (indicated by skill identifiers).

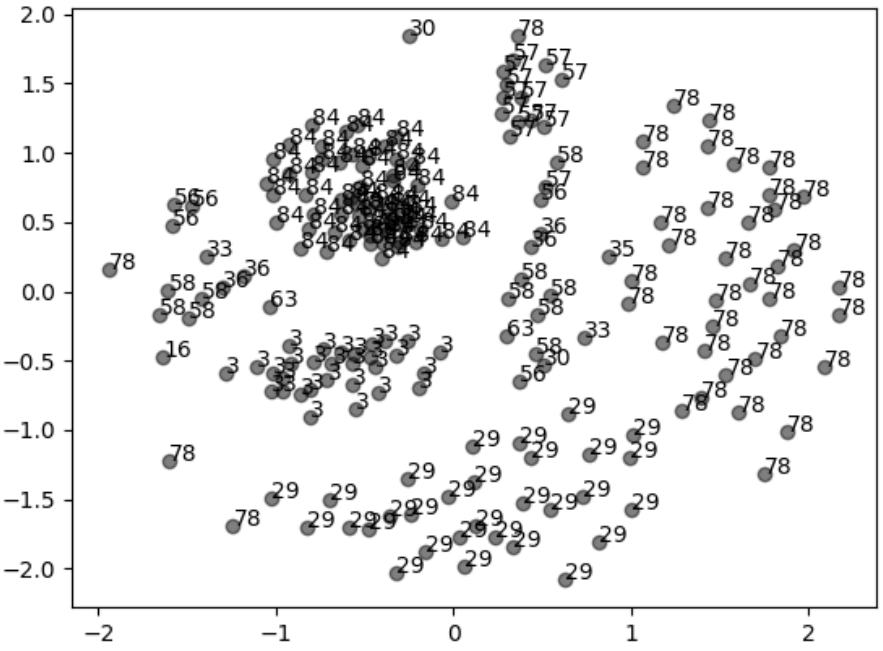

Figure 4.5: Visualization of the embedding of random selection of 200 questions by multidimensional scaling.

4.5 Conclusions

In this chapter, we investigated the dynamic low-rank matrix recovery problem with measurement induced dynamics from a practical perspective in the context of knowledge tracing. We presented a framework for tracking student knowledge in an ITS by using techniques from matrix factorization/embedding and RNNs. Our framework can track student knowledge without the concept/skill tag information required by other knowledge-tracing models, (e.g.,

DKT [76] and its variants). This avoids labor-intensive manual tagging. Taking advantage of additional latent question embeddings, our framework outperforms recent state-of-the-art knowledge-tracing models that only use RNNs, such as DKT. By constructing an embedding of the questions via matrix factorization in addition to skill-tagging information, our framework can fuse question-level and skill-level information, tracking student knowledge in a low-dimensional fused space. We also proposed and analyzed the necessity of pretraining the item embedding before training the dynamics model. The *DynEmb* framework is also flexible in that it can accommodate various matrix factorization techniques and dynamical models, which makes it a promising avenue for future research and development of algorithms for knowledge tracing.

However, for real-world implementation, several challenges remain regarding how to design a practical *DynEmb*-based system for knowledge tracing. For example, developing a method amenable to deployment in an online setting will require additional algorithmic improvements. Another challenge concerns how to incorporate additional sources of auxiliary information not considered here, such as question text or details about additional student interactions with an ITS (browsing history, textbook interactions, etc.) to exploit better all available information. We believe that the *DynEmb* framework provides a natural platform from which to address such challenges.

CHAPTER 5

RECOVERY GUARANTEES FOR LOW-RANK MATRIX RECOVERY FOR MEASUREMENT INDUCED DYNAMICS

In Chapter 4, we proposed the *DynEmb* framework for solving the dynamic low-rank matrix recovery problem with measurement induced dynamics from a practical perspective in the context of knowledge tracing. The dynamics model in *DynEmb* can be generally formulated as follows:

$$V^t = f\left(V^{t-1}, U, R_t\right), \quad t = 1, \ldots, d,$$

where the question matrix $U \in \mathbb{R}^{n_1 \times r}$ is fixed, the student matrix $V \in \mathbb{R}^{n_2 \times r}$ is changing over time and R is the interaction. Sequential models including RNNs, LSTMs, and memory networks are all included by this formulation. However, theoretical analysis for this general formulation is complicated even for the simplest RNNs. In this chapter, we first prototype a simple dynamic model from this general formulation and further reduce it to a static low-rank matrix recovery problem with a novel sampling ensemble. Then, we conduct some initial theoretical analysis of this static low-rank matrix recovery problem and present its implications for dynamic low-rank matrix recovery with measurement induced dynamics.

5.1 Motivation: A simple student learning dynamic model

In this section we present a particular measurement induced dynamics model using the student learning dynamic process as an example.

Suppose we have n students and m questions, and each of these students answer some of these m questions at time τ, hence generating an ordered sequence of total N observations $S := \{O_\tau := (R_\tau, i_\tau, k_\tau)\}_{\tau=1}^N$, where R_τ is the response of student $k_\tau \in [n]$ on question $i_\tau \in [m]$. Also let $S_{t,i,k} := \{O_\tau : \tau < t,\ \ i_\tau = i \ \text{ and } \ k_\tau = k \ \text{ for all } \ \tau \in [N]\}$. We assume

that the collection of questions is related to r abstract concepts. We let $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{n\times r}$ denote the question-concept matrix and the student-knowledge matrix. In particular, we use V^0 to denote the initial student-knowledge matrix before the learning process and V^τ to denote the student-knowledge matrix at time $\tau \in [N]$.

We are now ready to describe the learning process. At time $\tau \in [N]$, the response of student k_τ on question i_τ is

$$R_\tau = \left\langle U_{i_\tau}, V^\tau_{k_\tau} \right\rangle + w, \tag{5.1}$$

where w is the observation noise and $V^\tau_{k_\tau}$ is the knowledge vector of student k_τ at time τ.

We now consider several dynamic models for $V^\tau_{k_\tau}$. For simplicity we fix k_τ to a certain student and hence omit the subscript. We assume that the incremental change of V at time τ is always a scale of the corresponding question vector U_{i_τ}.

Context-free dynamic model

$$V^\tau = V^{\tau-1} + \alpha_{i_\tau} U_{i_\tau}. \tag{5.2}$$

In the context-free dynamic model, the incremental change of V at time τ is only related the corresponding measurement vector U_{i_τ}. The coefficient α_{i_τ} specifies how much change is applied after this interaction.

Context-aware dynamic model

$$V^\tau = V^{\tau-1} + \alpha(U, V^{\tau-1}, R)U_{i_\tau}, \tag{5.3}$$

This is a more realistic and practical model compared to the context-free dynamic model. The incremental change of V at time τ is determined by the current state $V^{\tau-1}$, interaction

R and measurement U_{i_τ}. One specific example of context-aware dynamic model is when the dynamics are determined by the similarity (inter product) of the current state (i.e., context) and the measurement vector:

$$V^\tau = V^{\tau-1} + \eta(R_\tau - \left\langle U_{i_\tau}, V^{\tau-1} \right\rangle) U_{i_\tau}. \tag{5.4}$$

Other commonly used context-aware dynamic models include, but are not limited to RNNs, LSTMs, GRUs, and RNNs with attention.

As we can see, in the context-free models the measurements can be reduced to a weighted linear sum of entries of low-rank matrix UU^T; while for the non context-free models, the measurements can be reduced to a weighted linear sum of entries of UU^T, $UU^T \circ UU^T$, ..., $UU^T \circ UU^T \circ \ldots \circ UU^t$, where $\circ$ is the element-wise product. The latter measurement model is significantly more complicated and far less amenable to analysis. Therefore, in this chapter we will focus on the context-free model. The context-free model might be less practically relevant, but it is easier to analyze, more intuitive and will provide some initial theoretical insights on dynamic low-rank matrix recovery with measurement induced dynamics.

As a first step towards analyzing this problem, we assume $\vec{\alpha} = (\alpha_1, \ldots, \alpha_m)$ is known and further that $\alpha_1 = \alpha_2 = \ldots = \alpha_m$. We want to estimate U and V. According to (5.2) at time τ, there is a rank-r matrix X^τ (see Figure 5.1):

$$X^\tau = UV^\tau = U(V^0)^T + UU^T \Gamma \mathbb{I}^\tau, \tag{5.5}$$

where Γ is a diagonal matrix with $\text{diag}(\Gamma) = \vec{\alpha}$, and $\mathbb{I}^\tau$ is the accumulation matrix at time τ with $[I^\tau]_{i,k} = |S_{\tau,i,k}|$.

Now our problem reduces to a mixture of matrix completion for $U(V^0)^T$ and matrix

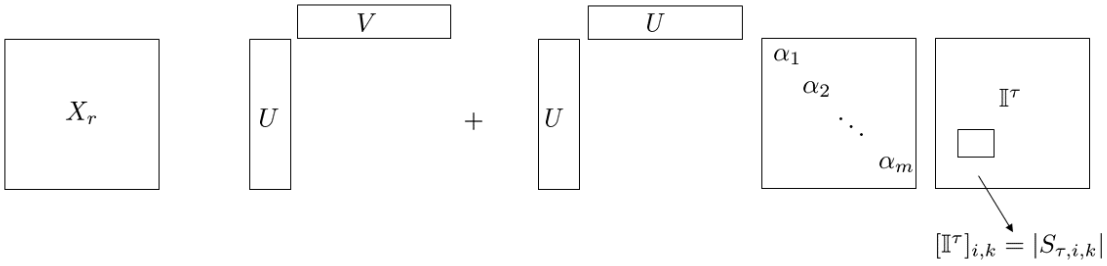

Figure 5.1: Low-rank matrix changing over time.

sensing for UU^T, i.e.,

$$R_\tau = \left\langle B_\tau, U(V^0)^T \right\rangle + \left\langle C_\tau, UU^T \right\rangle + w, \tag{5.6}$$

where B_τ is a sampling matrix and C_τ is a sensing matrix. For the sampling matrix B_τ, we have

$$[B_\tau]_{i,k} = \begin{cases} 1 & \text{if } i = i_\tau, k = k_\tau, \\ 0 & \text{otherwise.} \end{cases}$$

The sensing matrix $C_\tau = B_\tau \mathbb{I}_\tau^T \Gamma$ can be expressed as

$$C_\tau = \begin{bmatrix} \mathbf{0} \\ \mathbb{I}_{k_\tau}^\tau \circ \vec{\alpha} \\ \mathbf{0} \end{bmatrix}, \tag{5.7}$$

where only the i_τ^{th} row is non-zero and $\mathbb{I}_{k_\tau}^\tau$ is the k_τ^{th} column of $\mathbb{I}^\tau$.

Although previous works [32, 34, 35, 36] have achieved recovery guarantees for matrix completion on $U(V^0)^T$, there is no direct analysis dealing with measurements like C_τ. For the sake of analysis, by setting $V^0 = 0$ (this is conceptually valid when we don't have extra information on students), we extract mathematically an abstract problem: how to recover UU^T from $\{R_\tau\}_{\tau=1}^N$ according to (5.6) and (5.7). We can see that this problem is a low-rank matrix recovery problem. However the sensing matrices have the following properties:

- The sensing matrices are correlated instead of being independent as in prior literature.
- The sensing matrices are not fully dense as in matrix sensing and also not as sparse as in matrix completion.
- Compared to matrix completion or other structured random measurement problems, the sensing matrices are not drawn from some discrete orthonormal basis.

As our initial approach to this problem, we alternatively propose a novel Rademacher sub-sampling (RSS) measurement model and establish recovery guarantees for this model. First, this RSS model shares some common properties with (5.7). It is not fully dense as in conventional matrix sensing or sparse as in matrix completion. It will help reveal some initial insights on the recovery guarantees of low-rank matrix recovery with measurements as (5.7). Second, it is easier to analyze and also results in some interesting mathematical consequences by itself. In particular, there is not currently a unified treatment for both matrix sensing and matrix completion. The RSS measurement model accommodates both matrix sensing and matrix completion and provides an opportunity to develop a unified theory for matrix sensing and matrix completion. Finally, We leave establishing recovery guarantees from correlated measurements as our future work.

5.2 Revisiting matrix sensing and matrix completion

In recent years there has been a significant amount of progress in our understanding of how to recover a rank-r matrix from incomplete observations, even when the number of observations is much less than the number of entries in the matrix. (See [13] for an overview of this literature.) One general approach to this problem is to use nuclear norm minimization as a convex surrogate for the (non-convex) rank constraint [30, 31, 32, 36, 61, 51, 100]. Within this literature, there are a variety of approaches to measuring the underlying low-rank matrix that have been considered. In [30] and [31] the authors studied the general linear measurement case via the restricted isometric property (RIP). In their arguments, they show

that fully dense sensing matrices guarantee the matrix-RIP with high probability and hence enable uniform exact recovery (with high probability) through nuclear norm minimization. Another line of work analyzes the recovery guarantees when the sensing matrices are sparse, for example, matrix completion. For matrix completion, the sensing matrix is so sparse that the matrix-RIP does not hold. Researchers instead make incoherence assumptions on the unknown low-rank matrix ([32], [36]) and construct a dual certificate to show non-uniform recovery guarantees. Such methodological differences between matrix sensing and matrix completion are also observed when using non-convex approaches [54, 55, 56, 57, 58, 59, 60, 63, 101]. Given such methodological differences, one might question if there a unifying theory for matrix sensing, matrix completion, or a novel measurement scheme in which the sensing matrix is neither fully dense nor sparse? The answer is no. In this chapter we will develop a unifying low-rank matrix recovery theory for a novel unifying measurement scheme.

5.3 Problem formulation

First we define the following model of low-rank matrix recovery from linear measurements. Suppose a sequence of $n_1 \times n_2$ matrices $\{A_i\}_{i=1}^m$ are drawn i.i.d. from some set $\mathbb{A}$. Define the linear mapping $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$:

$$\mathcal{A}(X) = \begin{bmatrix} \langle A_1, X \rangle \\ \langle A_2, X \rangle \\ \vdots \\ \langle A_m, X \rangle \end{bmatrix}. \tag{5.8}$$

Also define the adjoint operator $\mathcal{A}^*$, such that for any $x \in \mathbb{R}^m$, $\mathcal{A}^*(x) = \sum_{i=1}^m x_i A_i$. Therefore the composite operator is $\mathcal{A}^*\mathcal{A}(X) = \sum_{i=1}^m \langle A_i, X \rangle A_i$ for any $X \in \mathbb{R}^{n_1 \times n_2}$.

Let X^* be an unknown rank-r matrix with SVD $X^* = U\Lambda V^T$. The goal is to recover X^* from $\mathcal{A}(X^*)$. When all A_i's contains only one non-zero entry, it is the measurement model

for matrix completion ([49]); and when all A_i's are dense and from certain i.i.d. distribution, it is the classical measurement model for matrix sensing, for example, Bernoulli/Gaussian ensemble ([30]).

In this chapter we consider the following measurement scheme, where each entry of $A_i \in \mathbb{R}^{n_1 \times n_2}$ follows distribution:

$$[A_i]_{k,l} = \begin{cases} \delta, & \text{with prob. } p/2 \\ -\delta, & \text{with prob. } p/2 \\ 0, & \text{with prob. } 1-p \end{cases}, \tag{5.9}$$

where $\delta = 1/\sqrt{pn_1n_2}$ is the normalization factor. One can verify that $\mathbb{E}[A_i]_{k,l} = 0$, $\mathbb{E}[A_i]_{k,l}^2 = 1/(n_1n_2)$ and $\mathbb{E}\left\|A_i\right\|_F^2 = 1$. We denote this as Rademacher sub-sampling (RSS) measurement ensemble.

As we can see A_i can be also reformulated as element-wise produce of two random matrix G_i and B_i, i.e., $A_i = \delta G_i \circ B_i$. The entries of G_i and B_i follow the following distribution

$$[G_i]_{k,l} = \begin{cases} 1, & \text{with prob. } 1/2 \\ -1, & \text{with prob. } 1/2 \end{cases} \quad \text{and} \quad [B_i]_{k,l} = \begin{cases} 0, & \text{with prob. } p \\ 1, & \text{with prob. } 1-p \end{cases}. \tag{5.10}$$

The above RSS measurement scheme accommodates both matrix sensing and matrix completion if one tunes the density parameter p in the range of $[1, \Theta(1/(n_1, n_2))]$, which yields a unifying theory for matrix sensing and matrix completion.

Now we consider the following constrained nuclear norm minimization program:

$$\begin{aligned} &\min \left\|X\right\|_*, \\ \text{s.t.} \quad &\mathcal{A}(X) = \mathcal{A}(X^*). \end{aligned} \tag{5.11}$$

Our goal is to prove that X^* is the unique solution to program (5.11) with high probability.

We have to point out that there are alternative approaches to recover the unknown low-rank matrix X^*, for example, alternating minimization [55] and gradient descent based on low-rank factorization [67]. Similar unifying analysis based these approaches for the RSS measurement scheme can also be established. However we do not pursue to conduct these alternative analysis in this thesis and leave them as future work.

5.4 Main results

5.4.1 Main theorem

Before we present our main theorem. We introduce the definitions for coherence and strong coherence parameters in matrix completion literature. For simplicity we use $n := n_1 + n_2$ from now.

Definition 5.4.1 (Coherence, [57])**.** A rank-r matrix $X \in \mathbb{R}^{n_1 \times n_2}$ with SVD $X^* = U\Sigma V^T$ is incoherent with parameter γ_0 if

$$\|U_i\|_2^2 \leq \frac{\gamma_0 r}{n_1} \quad \text{for any } i \in [n_1] \quad \text{and} \quad \|V_j\|_2^2 \leq \frac{\gamma_0 r}{n_2} \quad \text{for any } j \in [n_2].$$

.

One can verify that $\gamma_0 \in [1, \max(n_1, n_2)/r]$.

Definition 5.4.2 (Strong coherence, [36])**.** Suppose rank-r matrix $X^* = U\Sigma V^T$ and UV^T has a maximum entry bounded by $\gamma_1\sqrt{r/(n_1 n_2)}$ in absolute value for some positive γ_1, then we say X^* is γ_1 strong-incoherent.

One can also verify that $\gamma_1 \in [1, \sqrt{n_1 n_2}]$.

Now we present our main theorem in the following.

Theorem 5.4.3. *Assume $\mathcal{A}$ is the RSS measurement model and $n_1 = \Theta(n_2)$. For a fixed rank-r matrix X^* with SVD $X^* = U\Sigma V^T$, X^* is the exact solution to* (5.11) *with probability*

exceeding $1 - \Theta(1/n)$ *if the sample complexity satisfies*

$$m \gtrsim r \log^4 n \max \left\{ n_{\max}, \frac{1}{\sqrt{p}}, \frac{\gamma_0}{p n_{\min}}, \frac{\gamma_1^2}{n_1 n_2 p}, \frac{\gamma_1}{p\sqrt{n_1 n_2}}, \frac{\gamma_1 \sqrt{n_{\max}}}{\sqrt{p n_1 n_2}} \right\}. \tag{5.12}$$

Proof. See Section 5.5. □

Remark 5.4.4. The above results show that the sample complexity depends on: 1) the rank r and the dimensionalities n_1, n_2 of the unknown matrix; 2) the density of the sensing matrices p; 3) the coherence parameters γ_0, γ_1 of the unknown matrix.

Remark 5.4.5. The first factor $n_{\max}$ in the bracket in (5.12) indicates that the minimum sample complexity is $\Theta(n_{\max} r \log^4 n)$ regardless of γ_0, γ_1 and p. This matches the lower bound for matrix completion illustrated in [32]. However it has an additional logarithm factor compared to the matrix sensing results in [59]. The logarithm factor is due to the proof techniques we used.

Remark 5.4.6. It is also worth to point out that if $p \lesssim 1/(n_1 n_2)$, then the sample complexity will boost because the factor $\frac{1}{\sqrt{p}}$ in the bracket in (5.12) will dominate. Intuitively, when $p \lesssim 1/(n_1 n_2)$, then most of the sensing matrices will be all-zero matrices and hence the corresponding measurements are invalid. To guarantee unique recovery, one need to have way more sensing matrices to get enough valid measurements.

Remark 5.4.7. Note that from (5.12) we can roughly tell that if the sensing matrices are sparser (p is smaller) and the coherence parameters are larger (X^* is more coherent), then we need more measurements to guarantee the recovery of the unknown matrix X^*. In practice we usually cannot change γ_0, γ_1 and might have the freedom to choose p and m, then (5.12) provides theoretical guidelines for making trade-off between the density of sensing matrices and the number of measurements given the coherence parameters of the unknown low-rank matrix.

5.4.2 Consequences from Theorem 5.4.3

We also present some consequences from Theorem 5.4.3 based on different choices of sensing matrix density p in the following.

Matrix sensing

Corollary 5.4.8 (matrix sensing)**.** *Assume $\mathcal{A}$ is the RSS measurement model, $n_1 = \Theta(n_2)$, and $p = 1$. For a fixed rank-r matrix X^* with SVD $X^* = U\Sigma V^T$, X^* is the exact solution to* (5.11) *with probability exceeding $1 - \Theta(1/n)$ if the sample complexity satisfies*

$$m \gtrsim n_{\max} r \log^4 n$$

Proof. Plug in $p = 1$ to (5.12), use the fact that $\gamma_0 \in [1, \max(n_1, n_2)/r], \gamma_1 \in [1, \sqrt{n_1 n_2}]$, and simplify the expression. □

Remark 5.4.9. The above results are comparable to the results in [49] except an additional $\log^4 n$ factor. The logarithm factor is due to the matrix concentration inequality used in the proof.

Matrix completion

We also present our results for 'matrix completion', where the sensing matrix is super sparse, i.e., $p = \Theta(1/(n_1 n_2))$. This is not exactly the same as the conventional matrix completion since the sensing matrix may contains zero or more than one non-zero entries.

Corollary 5.4.10 (matrix completion)**.** *Assume $\mathcal{A}$ is the RSS measurement model, $n_1 = \Theta(n_2)$ and $p = \Theta(\frac{1}{n_1 n_2})$. For a fixed rank-$r$ matrix X^* with SVD $X^* = U\Sigma V^T$, X^* is the exact solution to* (5.11) *with probability exceeding $1 - \Theta(1/n)$ if the sample complexity satisfies*

$$m \gtrsim r \log^4 n \max\left\{\gamma_0 n_{\max}, \gamma_1 \sqrt{n_1 n_2}\right\}.$$

Proof. Plug in $p = \Theta(1/(n_1 n_2))$ to (5.12), use the fact that $\gamma_0 \in [1, \max(n_1, n_2)/r], \gamma_1 \in [1, \sqrt{n_1 n_2}]$, and simplify the expression. □

Remark 5.4.11. The results are similar as the classical matrix completion results in [36] except different dependency on coherence parameters and different order of logarithm factors of n. It is worth to point out that the dependence of sample complexity in the above corollary on γ_1 is better than that in [36]. Consider a rank-1 $n_0 \times n_0$ matrix containing only one none-zero entry, then $\gamma_0 = \gamma_1 = n_0$. According to [36], the sample complexity is $\Theta(n_0^3 \log^2 n_0)$, while our results show that the sample complexity is $\Theta(n_0^2 \log^4 n_0)$, which is closer to the lower bound $\Theta(n_0^2 \log n_0)$ illustrated in [32] (Theorem 1.7).

Sparse matrix sensing

Now we present a interesting consequence for 'sparse matrix sensing' from Theorem 5.5.4.

Corollary 5.4.12 (Sparse matrix sensing)**.** *Assume $\mathcal{A}$ is the RSS measurement model and $n_1 = \Theta(n_2)$. For a fixed rank-r matrix X^* with SVD $X^* = U\Sigma V^T$, X^* is the exact solution to* (5.11) *with probability exceeding $1 - \Theta(1/n)$ if the sample complexity satisfies*

$$m \gtrsim n_{\max} r \log^4 n,$$

provided that

$$p \gtrsim 1/\sqrt{n_1 n_2}.$$

Proof. Plug in $p \gtrsim 1/\sqrt{n_1 n_2}$ to (5.12), use the fact that $\gamma_0 \in [1, \max(n_1, n_2)/r], \gamma_1 \in [1, \sqrt{n_1 n_2}]$, and simplify the expression. □

Remark 5.4.13. The above corollary shows that one does not need fully dense sensing matrices, e.g., Gaussian/sub-Gaussian random matrices to achieve nearly optimal sample complexity guarantee even when recovering a extremely coherent matrix.

5.4.3 Implications for the measurement model in (5.7)

Rigorously proving similar recovery guarantees for the measurement model in (5.7) is challenging. We leave it as future work. However previous recovery guarantees cast some insights on it.

In a similar setting as in Section 5.1, assume we have m students and n questions, and each student answer k questions sequentially. The set of questions for each student are uniformly drawn from these n questions. Then the problem can be reduced as: can we recover a symmetric square rank-r matrix $X \in \mathbb{R}^{n\times n}$ from these mk measurements?

Ignoring the temporal dependency between measurements from a particular student, the recovery problem can be further reduced as: assume that each measurement matrix contains k non-zero entries, can we recover X from these mk independent measurements? Let $p = k/n^2$, then this exactly can be answered by the above main theorem:

$$mk \gtrsim r \log^4 n \max\left\{n, \frac{1}{\sqrt{p}}, \frac{\gamma_0}{pn}, \frac{\gamma_1^2}{n^2 p}, \frac{\gamma_1}{pn}, \frac{\gamma_1}{\sqrt{pn}}\right\}.$$

This provides us some guidelines to choose m and k in practice.

5.5 Proof outline of Theorem 5.4.3

We first introduce a preliminary theorem.

5.5.1 A preliminary theorem

Consider a rank-r matrix $X \in \mathbb{R}^{n_1\times n_2}$ with SVD $X = U\Sigma V^T$. Let u_k and v_k, $k \in [r]$, denote the left and right singular vectors respectively. We first introduce the orthogonal decomposition $\mathbb{R}^{n_1\times n_2} = T \bigoplus T^\perp$ where T is the linear space spanned by elements of the form $u_k y$ and $x v_k^T$, $1 \le k \le r$, where x and y are arbitrary, and $T^\perp$ is its orthogonal

complement. The orthogonal projection $\mathcal{P}_T$ onto T is given by

$$\mathcal{P}_T(Z) = P_U Z + Z P_V - P_U Z P_V,$$

where P_U and P_V are the orthogonal projections onto U and V respectively. Similarly the orthogonal projection on to $T^\perp$ is given by

$$\mathcal{P}_{T^\perp}(Z) = (\mathcal{I} - \mathcal{P}_T)(Z).$$

For simplicity, we denote $Z_T = \mathcal{P}_T(Z)$ and $Z_T^\perp = \mathcal{P}_{T^\perp}(Z)$.

Before presenting our main results, we recall that the linear measurement operator $\mathcal{A} = \{A_i\}_{i=1}^m$ and introduce several definitions.

Definition 5.5.1. The *coherence* parameter of $\mathcal{A}$ with respect to the linear space T is defined as

$$\mu(T) := \max_{i\in[m]} \|\mathcal{P}_T(A_i)\|_F^2 .$$

Definition 5.5.2. The *strong coherence* parameter of $\mathcal{A}$ with respect to a certain matrix $F \in \mathbb{R}^{n_1\times n_2}$ is defined as

$$\kappa(F) := \max_{i\in[m]} \langle F, A_i\rangle^2 / \|F\|_F^2 .$$

Definition 5.5.3. The *spectral bound* parameter of $\mathcal{A}$ is defined as

$$\nu(\mathcal{A}) := \max_{i\in[m]} \|A_i\| .$$

The above definition of coherence parameter $\mu(T)$ is different from cases where the sensing matrices are drawn from an orthonormal set such as matrix completion. In matrix completion, the maximum is taken over the entire measurement ensemble. However in the RSS model, the sensing matrices are drawn from an extremely large (technically $3^{n_1 n_2}$)

non-orthonormal set. In the case where $m \ll N$, defining the incoherence $\mu(T)$ over the entire measurement ensemble largely over-estimates $\mu(T)$, which will lead to loose recovery guarantees. Moreover calculating the maximum over $3^{n_1 n_2}$ sensing matrices takes exponential time, which is unpractical. To overcome these issues, we define the coherence $\mu(T)$ as the maximum of $\|\mathcal{P}_T(A_i)\|_F^2$ over $i \in [m]$. Note that $\mu(T)$ is different for each realization of the measurement operator $\mathcal{A}$ and can be extremely large (technically as large as $n_1 n_2$). So with this definition, the coherence in the RSS model can only be upper bounded probabilistically with respect to $\mathcal{A}$ (uniformly to all unknown matrices $X \in \mathbb{R}^{n_1 \times n_2}$), while the coherence parameter in matrix completion is bounded deterministically with respect to a fixed unknown matrix $X \in \mathbb{R}^{n_1 \times n_2}$. Similarly we can define and probabilistically bound $\kappa(F)$ and $\nu(\mathcal{A})$.

We are now ready to present our the following preliminary theorem.

Theorem 5.5.4. *Assume $\mathcal{A}$ is the RSS measurement model. For a fixed rank-r matrix X^* with SVD $X^* = U\Sigma V^T$, X^* is the exact solution to* (5.11) *with probability exceeding $1 - \Theta(1/n)$ if the sample complexity satisfies*

$$m \gtrsim n_1 n_2 \log^2 n \max\{\mu'(T), \nu(\mathcal{A})\sqrt{\kappa(UV^T)r}, r\kappa(UV^T)\},$$

where $\mu'(T) := \max\{\frac{1}{n_1 n_2}, \mu(T)\}$, $\nu(\mathcal{A})$ and $\kappa(UV^T)$ defined above.

Remark 5.5.5. The above results are similar as the classical matrix completion results [35, 36] in the sense that the sample complexity depends on some coherence parameters. However there is some notable differences. In our results, we additionally requires the spectral bound $\nu(\mathcal{A})$. And the parameters $\nu(\mathcal{A})$, $\mu(T)$ and $\kappa(UV^T)$ are all probabilistic with respective to measurement operator $\mathcal{A}$, while in classical matrix completion, the coherence parameters (including both coherence parameter and strong coherence parameter) are both deterministic with respective to the underlying matrix X^*.

5.5.2 Proof of Theorem 5.5.4

We present the proof outline for Theorem 5.5.4, which does not contain proofs for the supporting technical lemmas. Proofs for the supporting technical lemmas are included in Section 5.8. The proof is similar to that in [35] with minor modifications.

Proof. To prove X^* is the only solution to program (5.11) is equivalently to show that there is no nonzero matrix $\Delta \in \mathbb{R}^{n_1 \times n_2}$ such that

$$\mathcal{A}(\Delta) = 0 \quad \text{and}$$
$$\|X^* + \Delta\|_* \leq \|X^*\|_* .$$

Following the same framework from [35], we will show in the following two cases:

- *Case I:* $\|\Delta_T\|_F \geq C_0 \left\|\Delta_T^\perp\right\|_F$. We will show that $\mathcal{A}(\Delta) \neq 0$.
- *Case II:* $\|\Delta_T\|_F \leq C_0 \left\|\Delta_T^\perp\right\|_F$. We will show that if $\mathcal{A}(\Delta) = 0$ then $\|X^* + \Delta\|_* > \|X^*\|_*$.

The constant C_0 will be determined later.

Proof for case I

We will prove that $\|\mathcal{A}(\Delta)\| > 0$ with high probability. Note that

$$\|\mathcal{A}(\Delta)\| = \left\|\mathcal{A}(\Delta_T + \Delta_T^\perp)\right\| \geq \|\mathcal{A}(\Delta_T)\| - \left\|\mathcal{A}(\Delta_T^\perp)\right\| .$$

To prove that $\|\mathcal{A}(\Delta)\|$ it is sufficient to lower bound $\|\mathcal{A}(\Delta_T)\|$ and upper bound $\left\|\mathcal{A}(\Delta_T^\perp)\right\|$.

First we have the following quick upper bound on $\left\|\mathcal{A}(\Delta_T^\perp)\right\|$.

Lemma 5.5.6. *For the RSS measurement model, the following inequality*

$$\left\|\mathcal{A}(\Delta_T^\perp)\right\|^2 \leq 4m \left\|\Delta_T^\perp\right\|_F^2 \tag{5.13}$$

with probability at least $p_1 := 1 - \exp\left[-\frac{9mn_1n_2p}{5}\right]$.

Proof. See Section 5.8.2. □

Remark 5.5.7. Recht [36] gives a stronger bound by bounding the number of duplicates for the measurements (particularly for matrix completion): with probability at least $1 - n_2^{2-2\beta}$, $\left\|\mathcal{A}(\Delta_T^{\perp})\right\|^2 \leq \frac{\beta^2}{9}\log^2(n_2)\left\|\Delta_T^{\perp}\right\|_F^2$ for $\beta > 1$ and $n_2 \geq 9$. Such a tighter bound will eventually eliminate one of the two logarithm factors in the sample complexity.

Next we lower bound $\|\mathcal{A}(\Delta_T)\|^2$.

We introduce a concentration inequality on the measurement operator in the following lemma.

Lemma 5.5.8. *For the RSS model, the following concentration inequality*

$$\left\|\mathcal{P}_T - \frac{n_1n_2}{m}P_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T\right\|_2 \leq 1/2$$

holds with probability at least $p_2 := 1 - n\exp(\frac{-3m}{32n_1n_2\mu'(T)})$, *where* $\mu'(T) := \max\{\frac{1}{n_1n_2}, \mu(T)\}$.

Proof. See Section 5.8.3. □

Now we are ready to lower bound $\|\mathcal{A}(\Delta_T)\|^2$. We have

$$\begin{aligned}
\|\mathcal{A}(\Delta_T)\|^2 &= \langle \Delta_T, \mathcal{A}^*\mathcal{A}(\Delta_T)\rangle \\
&= \langle \Delta_T, \mathcal{P}_T\mathcal{A}^*\mathcal{A}(\Delta_T)\rangle \\
&= \frac{m}{n_1n_2}\langle \Delta_T, \Delta_T\rangle - \left\langle \Delta_T, \frac{m}{n_1n_2}\Delta_T - \mathcal{P}_T\mathcal{A}^*\mathcal{A}(\Delta_T)\right\rangle \\
&= \frac{m}{n_1n_2}\|\Delta_T\|_F^2 - \left\langle \Delta_T, \left(\frac{m}{n_1n_2}\mathcal{P}_T - \mathcal{P}_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T\right)\Delta_T\right\rangle \\
&\geq \frac{m}{n_1n_2}\|\Delta_T\|_F^2 - \left\|\frac{m}{n_1n_2}\mathcal{P}_T - \mathcal{P}_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T\right\|_2\|\Delta_T\|_F^2 \\
&\geq \frac{m}{2n_1n_2}\|\Delta_T\|_F^2 .
\end{aligned}$$

The last inequality holds due to Lemma 5.5.8. Therefore

$$\|\mathcal{A}(\Delta_T)\|^2 \geq \frac{m}{2n_1n_2}\|\Delta_T\|_F^2 \tag{5.14}$$

holds with probability exceeding p_2.

Combining (5.13) and (5.14), if we set $C_0 = 2\sqrt{n_1n_2}$

$$\|\mathcal{A}^*\mathcal{A}(\Delta)\|_F \geq \frac{m}{2n_1n_2}\|\Delta\|_F^2 - 2m\left\|\Delta_T^{\perp}\right\|_F^2 > \left(C_0^2\frac{m}{2n_1n_2} - 2m\right)\left\|\Delta_T^{\perp}\right\|_F^2 = 0.$$

The above inequality holds with probability exceeding

$$1-(1-p_1)-(1-p_2).$$

Remark 5.5.9. We can have tighter C_0 if we prove the alternative tighter bound on $\left\|\mathcal{A}(\Delta_T^{\perp})\right\|$.

Proof for case II

We want to prove if $\|\Delta_T\|_F \leq C_0\left\|\Delta_T^{\perp}\right\|_F$ and $\mathcal{A}(\Delta) = 0$ then $\|\Delta + X^*\|_* > \|X^*\|_*$. We'll prove this by constructing a certificate probabilistically similarly as [35].

Recall that $\|A\|_* = \sup_{\|B\|_2\leq 1}\langle B, A\rangle$. For $\mathcal{A}(\Delta) = 0$, pick $U_\perp$ and $V_\perp$ such that $[U, U_\perp]$

and $[V, V_\perp]$ are unitary matrices and that $\langle U_\perp V_\perp^T, \Delta_T^\perp \rangle = \left\|\Delta_T^\perp\right\|_*$. Then it follows that

$$\begin{aligned}
\|X^* + \Delta\|_* &= \sup_{\|M\|_2 \le 1} \langle M, X^* + \Delta \rangle \\
&\ge \langle UV^T + U_\perp V_\perp^T, X^* + \Delta \rangle \\
&\ge \|X^*\|_* + \langle UV^T + U_\perp V_\perp^T, \Delta \rangle \\
&= \|X^*\|_* + \langle UV^T + U_\perp V_\perp^T - Y, \Delta \rangle \quad \text{(Assume } \langle Y, \Delta \rangle = 0) \\
&= \|X^*\|_* + \langle UV^T - Y_T, \Delta_T \rangle + \langle U_\perp V_\perp^T - Y_T^\perp, \Delta_T^\perp \rangle \\
&= \|X^*\|_* + \langle UV^T - Y_T, \Delta_T \rangle + \langle U_\perp V_\perp^T, \Delta_T^\perp \rangle - \langle Y_T^\perp, \Delta_T^\perp \rangle \\
&\ge \|X^*\|_* - \left\|UV^T - Y_T\right\|_F \|\Delta_T\|_F + (1 - \left\|Y_T^\perp\right\|_2) \left\|\Delta_T^\perp\right\|_* \\
&> \|X^*\|_* - C_1 \|\Delta_T\|_F + (1 - C_2) \left\|\Delta_T^\perp\right\|_* \\
&\ge \|X^*\|_* - C_1 \|\Delta_T\|_F + (1 - C_2) \left\|\Delta_T^\perp\right\|_F \\
&\ge \|X^*\|_*,
\end{aligned}$$

where Y satisfies the following condition

$$\begin{aligned}
\langle Y, \Delta \rangle &= 0 \\
\left\|UV^T - Y_T\right\|_F &\le C_1 \\
\left\|Y_T^\perp\right\|_2 &< C_2,
\end{aligned}$$

and the constants satisfy

$$1 - C_2 \ge C_0 C_1.$$

The constants are $C_1 = 1/(4\sqrt{n_1 n_2})$ and $C_2 = 1/2$, which will be determined latter.

Now we want to prove that Y exists with high probability by the golfing scheme. Partition $1, 2, 3, \ldots, m$ into $h := \log_2(4\sqrt{4n_1 n_2 r})$ partitions of size q. Let Ω_j denote the jth partition. Define $W_0 = UV^T$ and set $Y_k = \sum_{j=1}^k \mathcal{R}_j(W_{j-1})$ and $W_k = UV^T - \mathcal{P}_T Y_k$

for $k = 1, \ldots, p$, where

$$R_j(X) = \frac{n_1 n_2}{q} \sum_{k \in \Omega_j} \langle A_k, X \rangle A_k.$$

Then Y_k is exactly the certificate Y we want.

It is easy to verify that $\langle Y_k, \Delta \rangle = 0$, since $\mathcal{A}(\Delta) = 0$.

Compute

$$\begin{aligned}
\|W_k\|_F &= \left\|UV^T - \mathcal{P}_T Y_k + \mathcal{P}_T Y_{k-1} - \mathcal{P}_T Y_{k-1}\right\|_F \\
&= \|W_{k-1} - \mathcal{P}_T \mathcal{R}_k(W_{k-1})\|_F \\
&= \|(\mathcal{P}_T - \mathcal{P}_T \mathcal{R}_k \mathcal{P}_T) W_{k-1}\|_F \\
&\leq \|\mathcal{P}_T - \mathcal{P}_T \mathcal{R}_k \mathcal{P}_T\|_2 \|W_{k-1}\|_F .
\end{aligned}$$

Then it follows that $\|W\|_k \leq 2^{-k} \|W_0\|_F = 2^{-k}\sqrt{r}$ with high probability provided that $\|\mathcal{P}_T - \mathcal{P}_T \mathcal{R}_j \mathcal{P}_T\|_2 \leq 1/2$ for all $k = 1, \ldots, h$ with high probability.

Taking the union bound, Lemma 5.5.8 implies that for all $k = 1, \ldots, h$,

$$\|\mathcal{P}_T - \mathcal{P}_T \mathcal{R}_j \mathcal{P}_T\|_2 \leq 1/2$$

holds with probability at least $p_3 := 1 - nh \exp(\frac{-3q}{32 n_1 n_2 \mu'(T)})$.

Plug in $h = \log_2(4\sqrt{n_1 n_2 r})$, we conclude that

$$\left\|UV^T - \mathcal{P}_T Y_k\right\|_F = \|W_h\|_F \leq \frac{1}{4\sqrt{n_1 n_2}}$$

with probability exceeding p_3.

Let $\left\|Y_T^{\perp}\right\|_2 = \|\mathcal{P}_{T^{\perp}} Y_h\|_2$. To upper bound $\left\|Y_T^{\perp}\right\|_2$ we have

$$\begin{aligned}
\left\|Y_T^{\perp}\right\|_2 &= \left\|\mathcal{P}_{T^{\perp}} \sum_{j=1}^{h} \mathcal{R}_j(W_{j-1})\right\|_2 \\
&\leq \sum_{j=1}^{h} \left\|\mathcal{P}_{T^{\perp}} \mathcal{R}_j(W_{j-1})\right\|_2 \\
&\leq \sum_{j=1}^{h} C_3 \left\|W_{j-1}\right\|_F \\
&\leq C_3 \sum_{j=1}^{h} 2^{-j} \left\|W_0\right\|_F \\
&\leq 2\sqrt{r} C_3 \\
&= 1/2.
\end{aligned}$$

The second inequality thanks to the following lemma.

Lemma 5.5.10. *For all* $j \in [h]$

$$\|\mathcal{P}_{T^{\perp}} \mathcal{R}_j(W_{j-1})\| \leq \frac{1}{4\sqrt{r}} \left\|W_{j-1}\right\|_F$$

holds with probability at least $1 - (1 - p_4) - (1 - p_5)$*, where*

$$p_4 = 1 - 2mh \exp\left(-\frac{q}{2n_1 n_2 \mu(T)}\right)$$

and

$$p_5 = 1 - hn \exp\left[-\min\left(\frac{q}{96 r n_1 n_2 \kappa(UV^T)}, \frac{q}{8\sqrt{\kappa(UV^T) r} n_1 n_2 \nu(\mathcal{A})}\right)\right].$$

Proof. See Section 5.8.4. □

The third inequality uses previously proven results $\left\|W_k\right\|_F \leq 1/2 \left\|W_{k-1}\right\|_F$ for all $k \in [h]$.

In sum we have

$$\left\|Y_T^{\perp}\right\| \leq 1/2.$$

with probability exceeding $1-(1-p_4)-(1-p_5)$.

Now the existence of the certificate Y assures that *CASE II* holds with probability exceeding $1-(1-p_3)(1-p_4)(1-p_5)$.

Combining Case I and Case II

So far we don't put any condition on the sample complexity m. Combining previous results for *CASE I* and *CASE II*, with $\mu(T), \mu(F)$ and $\nu(\mathcal{A})$ defined, we can say that X^* is the unique solution to program (5.11) with probability exceeding

$$p_a := 1 - \sum_{i=1}^{5}(1-p_i).$$

Note that p_2 will be absorbed into p_3. In sum, for a fixed rank-r matrix X^* with SVD $X^* = U\Sigma V^T$, X^* is the exact solution to (5.11) with probability exceeding (ignore the constants)

$$\begin{aligned}
&1-\Theta\left(\exp\left(-mn_1n_2p\right)\right)-\Theta\left(n\log(n_1n_2r)\exp\left(-\frac{m}{n_1n_2\mu'(T)\log(n_1n_2r)}\right)\right)\\
&-\Theta\left(m\log(n_1n_2r)\exp\left(-\frac{m}{n_1n_2\log(n_1n_2r)\mu(T)}\right)\right)\\
&-\Theta\left(n\log(n_1n_2r)\exp\left(-\frac{m}{n_1n_2r\log(n_1n_2r)}\min\left\{\frac{1}{\kappa(UV^T)},\frac{1}{\nu(\mathcal{A})\sqrt{\kappa(UV^T)}}\right\}\right)\right),
\end{aligned}$$

where $\mu'(T) := \max\left\{\frac{1}{n_1n_2}, \mu(T)\right\}$.

Note that $\Theta\left(\exp\left(-mn_1n_2p\right)\right) \lesssim \Theta(1/n)$and $\Theta(\log n_1n_2r) \approx \Theta(\log n)$, then the above

probability can be further simplified as

$$
\begin{aligned}
&1-\Theta\left(n\log n\exp\left(-\frac{m}{n_1n_2\log n\mu'(T)}\right)\right)\\
&-\Theta\left(m\log n\exp\left(-\frac{m}{n_1n_2\log n\mu(T)}\right)\right)\\
&-\Theta\left(n\log n\exp\left(-\min\left\{\frac{m}{n_1n_2\kappa(UV^T)r\log n},\frac{m}{n_1n_2\log n\nu(\mathcal{A})\sqrt{\kappa(UV^T)r}}\right\}\right)\right).
\end{aligned}
$$

Note that $m \gg n$, and plug in (5.5.4) one can verify that the resulting probability is $1-\Theta(1/n)$. Therefore we complete the proof. □

5.5.3 Bounding probabilistic coherence parameters

The parameters $\mu(T), \kappa(UV^T)$ and $\nu(\mathcal{A})$ play key roles in Theorem 5.5.4. One of the main differences between our RSS ensemble and matrix completion is that even for a fixed unknown low-rank matrix, these parameters are extremely large with non-zero probability. However we can bound them with large probability as showed in the following lemmas.

The next three lemmas state the relationship between parameters $\mu(T)$, $\kappa(UV^T)$ and $\nu(\mathcal{A})$ and the classical deterministic coherence parameters γ_0 and γ_1 in classical matrix completion literature.

Lemma 5.5.11. *Assume $\mathcal{A}$ is the RSS measurement model. For some universal positive constant C we have*

$$
\mathbb{P}\left\{\mu(T)\geq\frac{C\log n}{n_1n_2p}\max\left(\frac{\gamma_0r\sqrt{\log n}}{n_{\min}},\frac{rpn_{\max}}{\log n}\right)\right\}\leq\frac{6}{n}. \tag{5.15}
$$

Proof. See Section 5.8.6. □

Lemma 5.5.12. *Assume $\mathcal{A}$ is the RSS measurement model. For some universal positive constant C we have*

$$
\mathbb{P}\left\{\kappa(UV^T)\geq\frac{C\log n}{n_1n_2p}\max\left(\frac{\log n}{n_1n_2}\gamma_1^2,p\right)\right\}\leq\frac{2}{n}. \tag{5.16}
$$

Proof. See Section 5.8.7. □

Lemma 5.5.13. *Assume $\mathcal{A}$ is the RSS measurement model. For some universal positive constant C we have*

$$\mathbb{P}\left\{\nu(\mathcal{A}) \geq C\delta \max\left(\sqrt{\log n}, \sqrt{n_{\max} p}\right)\right\} \leq \frac{3}{n},$$

where C is some universal positive constant.

Proof. See Section 5.8.8. □

5.5.4 Proof of Theorem 5.4.3

Proof. The proof is based on Lemma 5.5.11, 5.5.12 and 5.5.13 and Theorem 5.5.4.

First according to 5.5.12, 5.5.13 and the union bound argument, for some constant C, we have

$$\nu(\mathcal{A})\sqrt{\kappa(UV^T)r} \leq C \max\left\{\frac{\gamma_1\sqrt{r}\log^{3/2} n}{p(n_1n_2)^{3/2}}, \frac{\gamma_1\sqrt{r}\log n\sqrt{n_{\max}}}{p^{1/2}(n_1n_2)^{3/2}}, \frac{\sqrt{r}\log n}{p^{1/2}n_1n_2}, \frac{\sqrt{rn_{\max}\log n}}{n_1n_2}\right\} \tag{5.17}$$

with probability exceeding $1 - \frac{5}{n}$.

Let $m_0 := n_1n_2\log^2 n \max\{\mu'(T), \nu(\mathcal{A})\sqrt{\kappa(UV^T)r}, r\kappa(UV^T)\}$. Then plug in (5.15), (5.16) and (5.17) to the sample complexity condition in Theorem 5.5.4, and apply the standard union bound, we have $m \gtrsim m_0$ holds with probability exceeding $1 - \frac{13}{n}$ if for some constant C,

$$m \geq Cr\log^4 n \max\left\{n_{\max}, \frac{1}{\sqrt{p}}, \frac{\gamma_0}{pn_{\min}}, \frac{\gamma_1^2}{n_1n_2p}, \frac{\gamma_1}{p\sqrt{n_1n_2}}, \frac{\gamma_1\sqrt{n_{\max}}}{\sqrt{pn_1n_2}}\right\}.$$

□

5.6 Simulations

To demonstrate our main results, we conducted a series of numerical experiments for a variety of matrix coherence γ, density of measurement matrices p and number of measurements m. For each combination of (γ, p, m), we run the following single simulation N times. For one single simulation, we first randomly generate underlying low-rank matrix X, an $n \times n$ matrix of rank r according to the matrix coherence γ. Second, we generate a series of i.i.d. $n \times n$ measurement matrices $\{A_i\}_{i=1}^m$ according to (5.9). Finally, we generate measurements $\mathcal{A}(X)$ according to (5.8).

There are now several off-the-shelf algorithms that solve the convex program (5.11). For example, reformulate program (5.11) as a semi-definite program and use interior-point method to solve it(see [1]). Another popular algorithm is singular value thresholding (SVT, see [102]), and some variants based on SVT (see [103, 104]). In our experiments we use the linearized Bregman method for low-rank matrix recovery (see [104], sample code provided online). We use the relative error $e := \left\|\hat{X} - X\right\|_F / \|X\|_F$, where $\hat{X}$ is the solution returned by the solver. We declare X to be recovered if the solution returned by the solver satisfied $e < 0.1$.

For the first experiment, we fix the dimension of matrix X as $n = 100$ and the rank as $r = 3$. We consider two extreme cases regarding the matrix coherence γ: incoherent matrix X_{incoh}, which is generated by multiplying two i.i.d. Gaussian matrices $U \in \mathbb{R}^{n\times r}$ and $V \in \mathbb{R}^{r\times n}$; coherent matrix X_{coh}, which is a diagonal matrix whose diagonal contains r ones uniformly. Then we generate the following rank-r matrix with parameter coh: $X = [cohU_{coh} + (1 - coh)U_{incoh}][cohV_{coh} + (1 - coh)V_{incoh}]^T$, and normalize X. For each p we generate a series of $m \in [3nr, 3n^2]$. For each combination (p, m) we run $N = 12$ simulations and compute the recovery success rate for that combination. We show the recovery success rates in Figure 5.2 under different combinations of p and m. As we can see from Figure 5.2, location of phase transition is related to the coh parameter. The more

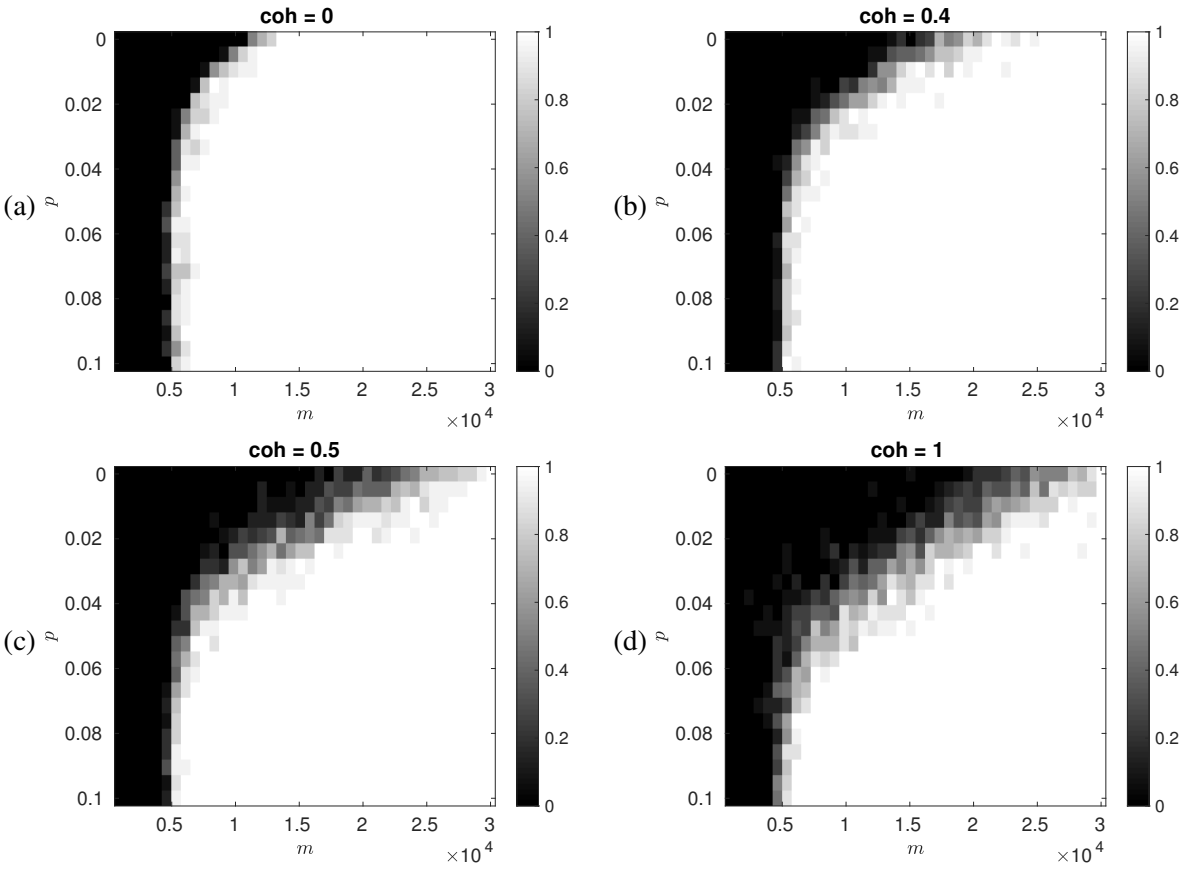

Figure 5.2: Phase transitions when matrices are of different coherences coh.

coherent the matrix is, the higher sample complexity where phase transition happens.

For the second experiment, the setup is the same as the first experiment. We further find the minimum m_0 that successfully recover X, i.e., the sample complexity to recover the matrix with success rate greater than 0.9. Our purpose is to investigate how the sample complexity changes when the measurement matrices are getting denser and denser for low-rank matrices with various matrix coherence parameters. The results are present in Figure 5.3. As we can see from Figure 5.3 the plots align together when p greater than $\Theta(1/n)$ and start to diverge when p is smaller than $\Theta(1/n)$, which validates the results in Corollary 5.4.12.

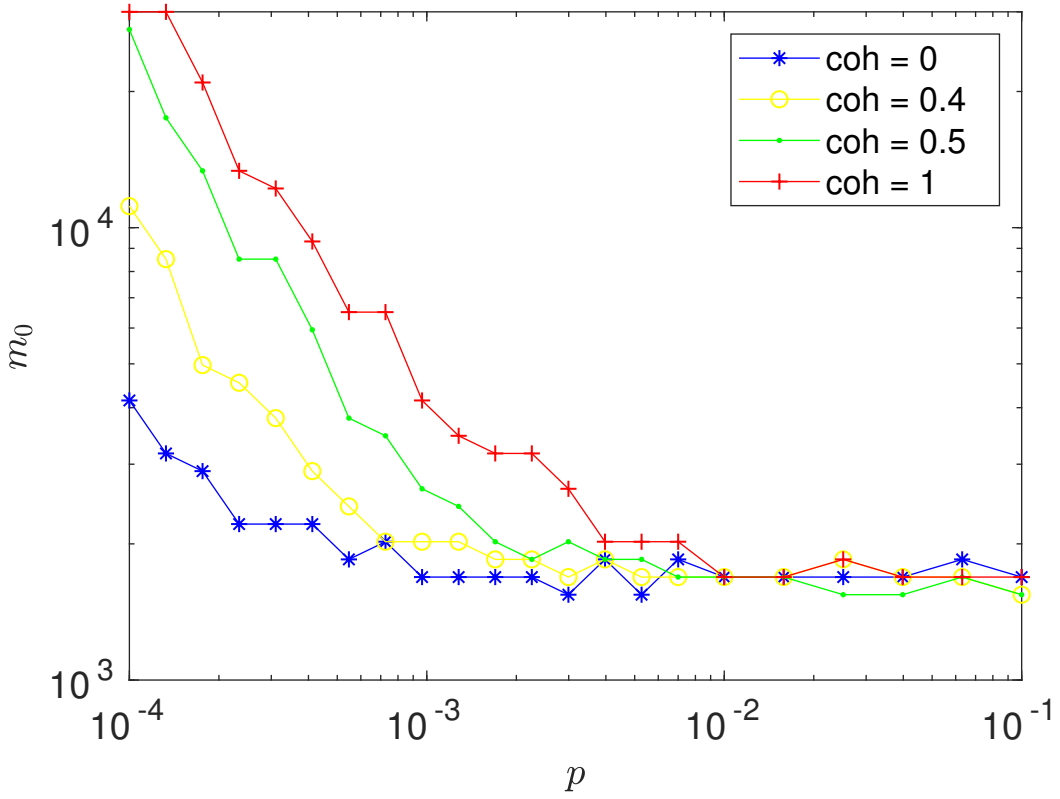

Figure 5.3: Sample complexity under different measurement densities p and coherences coh.

5.7 Conclusions

In this chapter we propose a novel RSS measurement model for low-rank matrix recovery and establish recovery guarantees for it. The recovery guarantees accommodate both matrix sensing and matrix completion and curve how the classical incoherence property in matrix completion analysis comes into play when the sensing matrix gets sparser starting from the fully dense case. To our surprise, our analysis shows that one do not need full dense sensing matrix to recover with high probability even when the unknown low-rank matrix is extremely incoherent.

There are still some further works can be done to improve the theory. For example, one can potentially eliminate the extra logarithm factor in the sample complexity in the fully dense case. Another possibility is to develop lower bound on the recovery guarantees. Alternatively one can consider a measurement scheme where each sensing matrix contains a fixed number of non-zeros (± 1) entries and the non-zero locations are uniformly sampled. However such measurement ensemble is way more complicated, since entries in the sensing matrix are correlated.

At last, we want to point out the theory does not fully solve the starting case when the sensing matrix is the one as (5.7). We will leave it as future work.

5.8 Technical proof details

5.8.1 Preliminary inequalities

Before we proceed our proof, we first introduce the following Non-commutative matrix Bernstein Inequality.

Theorem 5.8.1 ([35]). *Let $X_1, \ldots, X_L$ be independent zero-mean random matrices of dimension $d_1 \times d_2$. Suppose $\rho_k^2 = \max\{\|\mathbb{E}[X_k X_k^*]\|_2, \|\mathbb{E}[X_k^* X_k]\|_2\}$ and $\|X_k\|_2 \leq M$*

almost surely for all k. Then for any $\tau > 0$.

$$\mathbb{P}\left[\left\|\sum_{k=1}^{L} X_k\right\|_2 > \tau\right] \leq (d_1 + d_2)\exp\left(\frac{-\tau^2/2}{\sum_{k=1}^{L}\rho_k^2 + M\tau/3}\right).$$

When $\tau \leq \sum_{i=1}^{m}\rho_i^2/M$, the bound can be further reduced to $(d_1+d_2)\exp\left(\frac{-3\tau^2/8}{\sum_{k=1}^{L}\rho_k^2}\right)$.

Next We introduce the following Hanson-Wright inequality.

Theorem 5.8.2 ([105]). *Let $X = (X_1, \ldots, X_n) \in \mathbb{R}^n$ be a random vector with independent components X_i which satisfies $\mathbb{E}X_i = 0$ and $\|X_i\|_{\psi_2} \leq K$ for all $i \in [n]$. Let A be an $n \times n$ matrix. Then for every $t > 0$,*

$$\mathbb{P}\left\{|X^T AX - \mathbb{E}X^T AX| > t\right\} \leq 2\exp\left[-c\min\left(\frac{t^2}{K^4\|A\|_F^2}, \frac{t}{K^2\|A\|_2}\right)\right],$$

where c is a positive constant.

We also introduce some bounds on the tail probabilities of some random variables.

Let $a_1, a_2, \ldots, a_n$ be reals in $(0, 1]$. Let $X_1, X_2, \ldots, X_n$ be independent Bernoulli trials with $\mathbb{E}[X_i] = p_i$. Define random variable $X = \sum_{i=1}^{n} a_i X_i$. Then $\mathbb{E}[X] = \sum_{i=1}^{n} a_i p_i$. We have the following Chernoff-type bound on the deviations of X above its mean.

Theorem 5.8.3 ([106]). *Let $\delta > 0$, $a_{\max} = \max_{i\in[n]} a_i$ and $m = \mathbb{E}[X]/a_{\max} \geq 0$. Then*

$$\mathbb{P}\left\{X \geq (1+\delta)\mathbb{E}[X]\right\} \leq \left[\frac{e^\delta}{(1+\delta)^{1+\delta}}\right]^m.$$

Similarly for $\gamma \in (0, 1]$,

$$\mathbb{P}\left\{X \leq (1-\gamma)\mathbb{E}[X]\right\} \leq \left[\frac{e^\gamma}{(1+\gamma)^{1+\gamma}}\right]^m.$$

More convenient bounds are:

$$\mathbb{P}\left\{X \geq (1+\delta)\mathbb{E}[X]\right\} \leq e^{-\frac{\delta^2 m}{2+\delta}},$$

and

$$\mathbb{P}\left\{X \leq (1-\gamma)\mathbb{E}[X]\right\} \leq e^{-\frac{\gamma^2 m}{2}}.$$

We use these inequalities substantially in the proofs of our technique lemmas.

5.8.2 Proof of Lemma 5.5.6

Proof. For a particular i, $\|A_i\|_F^2 \sim \delta^2$ Binomial$(n_1 n_2, p)$. Therefore $\sum_{i=1}^m \|A_i\|_F^2 \sim \delta^2$ Binomial$(n_1 n_2 m, p)$ Note that $\mathbb{E}\sum_{i=1}^m \|A_i\|_F^2 = m$. According to Theorem 5.8.3, we have

$$\mathbb{P}\left[\sum_{i=1}^m \|A_i\|_F^2 \geq 4m\right] \leq \exp\left[-\frac{9mn_1n_2p}{5}\right].$$

Since $\left\|\mathcal{A}(\Delta_T^\perp)\right\|^2 \leq \sum_{i=1}^m \|A_i\|_F^2 \left\|\Delta_T^\perp\right\|_F^2$, we complete the proof. □

5.8.3 Proof of Lemma 5.5.8

Proof. Define an operator $\mathcal{G}_i$ which maps Z to $\langle \mathcal{P}_T(A_i), Z\rangle \mathcal{P}_T(A_i)$. Then $\mathcal{P}_T\mathcal{A}^*\mathcal{A}P_T = \sum_{i=1}^m \mathcal{G}_i$. Also Let $\mathcal{T}_i(Z) = \langle A_i, Z\rangle A_i$.

First compute $\mathbb{E}\mathcal{T}_i(Z) = \mathbb{E}\langle A_i, Z\rangle A_i$. For a particular entry of $\mathbb{E}\mathcal{T}_i(Z)$ we have

$$\begin{aligned}
[\mathbb{E}\mathcal{T}_i(Z)]_{k,l} &= \mathbb{E}\langle Z, A_i\rangle [A_i]_{k,l} \\
&= \mathbb{E}\sum_{(g,h)\neq(k,l)} [Z]_{g,h}[A_i]_{g,h}[A_i]_{k,l} + \mathbb{E}[Z]_{k,l}[A_i]_{k,l}[A_i]_{k,l} \\
&= 0 + \frac{1}{n_1 n_2}[Z]_{k,l}.
\end{aligned}$$

So $\mathbb{E}\mathcal{T}_i(Z) = \frac{1}{n_1 n_2} Z$, and hence $\mathbb{E}\mathcal{P}_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T(Z) = \frac{m}{n_1 n_2}\mathcal{P}_T(Z)$.

Consider the following random variable

$$\|\mathbb{E}\mathcal{P}_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T - \mathcal{P}_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T\|_2 = \left\|\sum_{i=1}^m (\mathcal{G}_i - \mathbb{E}\mathcal{G}_i)\right\|_2,$$

which is the spectral norm of a sum of i.i.d zero-mean random matrices. This means we

might apply the matrix Bernstein inequality.

First bound $\|\mathcal{G}_i - \mathbb{E}\mathcal{G}_i\|_2$ in the following:

$$\|\mathcal{G}_i - \mathbb{E}\mathcal{G}_i\|_2 = \max\{\|\mathcal{G}_i\|_2, \|\mathbb{E}\mathcal{G}_i\|_2\}.$$

For $\|\mathcal{G}_i\|_2$ we have

$$\begin{aligned}
\|\mathcal{G}_i\|_2 &= \sup_{X \in \mathbb{R}^{n_1 \times n_2}, X \neq 0} \frac{\|\mathcal{G}_i(X)\|_F}{\|X\|_F} \\
&= \sup_{X \in \mathbb{R}^{n_1 \times n_2}, X \neq 0} \frac{\|\langle \mathcal{P}_T(A_i), X \rangle \mathcal{P}_T(A_i)\|_F}{\|X\|_F} \\
&= \|\mathcal{P}_T(A_i)\|_F^2 .
\end{aligned}$$

What we want to bound is $\max_{i \in [m]} \|\mathcal{P}_T(A_i)\|_F^2$, which is exactly the definition of coherence. We assume that the coherence parameter for T and $\mathcal{A}$ is $\mu(T)$, then

$$\|\mathcal{G}_i\|_2 \leq \mu(T)$$

for all $i \in [m]$.

For $\|\mathbb{E}\mathcal{G}_i\|_2$ we have

$$\begin{aligned}
\|\mathbb{E}\mathcal{G}_i\|_2 &= \left\| \frac{1}{n_1 n_2} \mathcal{P}_T \right\|_2 \\
&= \frac{1}{n_1 n_2}.
\end{aligned}$$

Obviously combining the two bounds above we have $\|\mathcal{G}_i - \mathbb{E}\mathcal{G}_i\|_2 \leq \max\{\frac{1}{n_1 n_2}, \mu(T)\} =: \mu'(T)$.

Second bound $\|\mathbb{E}[(\mathcal{G}_i - \mathbb{E}\mathcal{G}_i)^2]\|_2$ in the following:

$$\begin{aligned}
\left\|\mathbb{E}[(\mathcal{G}_i - \mathbb{E}\mathcal{G}_i)^2]\right\|_2 &= \left\|\mathbb{E}\mathcal{G}_i^2 - (\mathbb{E}\mathcal{G}_i)^2\right\|_2 \\
&= \left\|\mathbb{E}[\|\mathcal{P}_T(A_i)\|_F^2 \mathcal{G}_i] - (\mathbb{E}\mathcal{G}_i)^2\right\|_2 \\
&\leq \max\left\{\left\|\mathbb{E}[\|\mathcal{P}_T(A_i)\|_F^2 \mathcal{G}_i]\right\|_2, \left\|(\mathbb{E}\mathcal{G}_i)^2\right\|_2\right\} \\
&\leq \max\left\{\|\mu(T)\mathbb{E}[\mathcal{G}_i]\|_2, \left\|(\mathbb{E}\mathcal{G}_i)^2\right\|_2\right\} \\
&= \max\left\{\frac{\mu(T)}{n_1 n_2}, \frac{1}{n_1^2 n_2^2}\right\}.
\end{aligned}$$

Now if $\tau \leq \frac{m}{n_1 n_2}$, we can substitute the following parameters

$$\sum_{i=1}^{m} \rho_i^2 = \frac{m\mu'(T)}{n_1 n_2},$$

$$M = \mu'(T),$$

to the matrix Bernstein inequality in Theorem 5.8.1, which leads to the following inequality

$$\|\mathbb{E}\mathcal{P}_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T - \mathcal{P}_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T\|_2 \leq \tau$$

holds with probability at least $1 - n\exp(-3\tau^2/8(\sum_{i=1}^{m}\rho_i^2))$, when $\tau \leq \frac{m}{n_1 n_2}$

Let $\tau = \frac{1}{2}\frac{m}{n_1 n_2}$ we have

$$\left\|\mathcal{P}_T - \frac{n_1 n_2}{m} P_T\mathcal{A}^*\mathcal{A}\mathcal{P}_T\right\|_2 \leq 1/2$$

with probability at least $p_2 := 1 - n\exp(\frac{-3m}{32 n_1 n_2 \mu'(T)})$.

□

5.8.4 Proof of Lemma 5.5.10

Proof. First we introduce the following lemma:

Lemma 5.8.4. *Let $F \in T$*

$$\|\mathcal{P}_{T^\perp}\mathcal{R}_j(F)\| \leq \frac{1}{4\sqrt{r}}\|F\|_F$$

holds with probability at least

$$1 - n\exp\left[-\min\left(\frac{q}{96rn_1n_2\kappa(F)}, \frac{q}{8\sqrt{\kappa(F)r}n_1n_2\nu(\mathcal{A})}\right)\right].$$

Also we introduce a lemma similar as Lemma 10 in [35] on $\mu\left((\mathcal{I} - \mathcal{P}_T\mathcal{R}\mathcal{P}_T)F\right)$ given $\mu(F)$:

Lemma 5.8.5. *Let $F \in T$. Then*

$$\mathbb{P}\left[\kappa\left((\mathcal{I} - \mathcal{P}_T\mathcal{R}\mathcal{P}_T)F\right) > \kappa(F)/2\right] \leq 2m\exp\left(-\frac{q}{2n_1n_2\mu(T)}\right).$$

Based on Lemma 5.8.5 and the standard union bound, we conclude that for all $j = 0, \ldots, h, \kappa(W_j) \leq \kappa(W_0) = \kappa(UV)$ with probability exceeding

$$p_4 := 1 - 2hm\exp\left(-\frac{q}{2n_1n_2\mu(T)}\right).$$

Now according to Lemma 5.8.4, if for all $j \in [h], \kappa(W_j) \leq \kappa(UV^T)$ then

$$\|\mathcal{P}_{T^\perp}\mathcal{R}_j(W_j)\| \leq \frac{1}{4\sqrt{r}}\|W_j\|_F$$

holds with probability at least

$$p_5 := 1 - hn\exp\left[-\min\left(\frac{q}{96rn_1n_2\kappa(UV^T)}, \frac{q}{8\sqrt{\kappa(UV^T)r}n_1n_2\nu(\mathcal{A})}\right)\right].$$

Now we complete the proof. □

5.8.5 Proofs of supporting lemmas for Lemma 5.5.10

Proof of Lemma 5.8.4

Proof. It is suffice to treat the case where $\|F\|_F = 1$. Set

$$X_a = \frac{n_1 n_2}{q} \mathcal{P}_{T^\perp} A_a \langle F, A_a \rangle .$$

Then $\mathcal{P}_{T^\perp} \mathcal{R}_j(F) = \sum_{a=1}^q X_a$, and

$$\mathbb{E}[X_a] = \frac{n_1 n_2}{q} \mathcal{P}_T^\perp \frac{1}{n_1 n_2} F = 0.$$

To apply the matrix Bernstein, use the fact that $\left\| P_T^\perp A_a \right\| \leq \|A_a\|$ and we compute

$$\begin{aligned} \left\| \mathbb{E}[X_a X_a^T] \right\| &= \frac{n_1^2 n_2^2}{q^2} \left\| \mathbb{E} \langle F, A_a \rangle^2 P_T^\perp(A_a) [P_T^\perp(A_a)]^T \right\| \\ &\leq \frac{n_1^2 n_2^2}{q^2} \left\| \mathbb{E} \langle F, A_a \rangle^2 (A_a) [(A_a)]^T \right\| . \end{aligned}$$

Let $H = \langle F, A \rangle^2 AA^T$ (neglecting a for simple notation), then

$$H_{kl} = \left(\sum_{i,j} F_{i,j} A_{i,j} \right)^2 \left(\sum_{h=1}^{n_2} A_{kh} A_{lh} \right) .$$

When $k = l$ then

$$\begin{aligned} \mathbb{E}[H_{kk}] &= \mathbb{E} \left(\sum_{i,j} F_{i,j} A_{i,j} \right)^2 \left(\sum_{h=1}^{n_2} A_{kh} A_{kh} \right) \\ &= \mathbb{E} \sum_{h=1}^{n_2} A_{kh} A_{kh} A_{kh}^2 F_{kh}^2 \\ &= \sum_{h=1}^{n_2} F_{kh}^2 \mathbb{E} A_{kh}^4 \\ &= p\delta^4 \sum_{h=1}^{n_2} F_{kh}^2 . \end{aligned}$$

On the other hand

$$\begin{aligned}\mathbb{E}[H_{kk}] &= \mathbb{E}\left(\sum_{i,j} F_{i,j}A_{i,j}\right)^2 \left(\sum_{h=1}^{n_2} A_{kh}A_{kh}\right) \\ &\leq \kappa(F)\mathbb{E}\left(\sum_{h=1}^{n_2} A_{kh}A_{kh}\right) \\ &= \frac{\kappa(F)}{n_1 n_2}.\end{aligned}$$

When $k \neq l$ then

$$\begin{aligned}\mathbb{E}[H_{kl}] &= \mathbb{E}\left(\sum_{i,j} F_{i,j}A_{i,j}\right)^2 \left(\sum_{h=1}^{n_2} A_{kh}A_{lh}\right) \\ &= 2\mathbb{E}\sum_{h=1}^{n_2} A_{lh}^2 A_{kh}^2 F_{kh}F_{lh} \\ &= 2\sum_{h=1}^{n_2} F_{kh}F_{lh}\mathbb{E}A_{kh}^2 A_{lh}^2 \\ &= 2p^2\delta^4 \sum_{h=1}^{n_2} F_{kh}F_{lh}.\end{aligned}$$

So the spectral norm of $\mathbb{E}H$ can be bounded as

$$\begin{aligned}\|\mathbb{E}H\| &= \delta^4 \left\|2p^2 FF^T + (p-2p^2)\text{diag}(FF^T)\right\| \\ &\leq 2p^2\delta^4 \left\|FF^T\right\| + p\delta^4 \left\|\text{diag}(FF^T)\right\| \\ &\leq 2p^2\delta^4 + p\delta^4 \left\|\text{diag}(FF^T)\right\| \\ &\leq 2p^2\delta^4 + \frac{\kappa(F)}{n_1 n_2} \\ &= \frac{2}{n_1^2 n_2^2} + \frac{\kappa(F)}{n_1 n_2} \\ &\leq \frac{3\kappa(F)}{n_1 n_2}.\end{aligned}$$

The first inequality uses triangle inequality. The second inequality uses the fact that $\left\|FF^T\right\| \leq \|F\|_F^2$. The last use the fact that $\kappa(F) \geq \frac{1}{n_1 n_2}$ with overwhelming probability

(larger than $1 - \Theta(1/n)$) regardless of p (see Lemma 5.5.12). So we have

$$\left\|\mathbb{E}[X_a X_a^T]\right\| \leq \frac{3n_1n_2\kappa(F)}{q^2}$$

and similarly

$$\left\|\mathbb{E}[X_a^T X_a]\right\| \leq \frac{3n_1n_2\kappa(F)}{q^2}.$$

So $V_0 =: q\max\{\left\|\mathbb{E}[X_a^T X_a]\right\|, \left\|\mathbb{E}[X_a X_a^T]\right\|\} = \frac{3n_1n_2\kappa(F)}{q}$.

Next compute

$$\begin{aligned}
\|X_a\| &= \left\| \frac{n_1n_2}{q} \mathcal{P}_{T^\perp} A_a \langle F, A_a \rangle \right\| \\
&= |\langle F, A_a \rangle| \frac{n_1n_2}{q} \|\mathcal{P}_{T^\perp} A_a\| \\
&\leq |\langle F, A_a \rangle| \frac{n_1n_2}{q} \|A_a\| \\
&\leq \sqrt{\kappa(F)} \frac{n_1n_2}{q} \|A_a\| \\
&\leq \sqrt{\kappa(F)} \frac{n_1n_2}{q} \nu(\mathcal{A}).
\end{aligned}$$

So $b = \sqrt{\kappa(F)}\frac{n_1n_2}{q}\nu(\mathcal{A})$. Applying Matrix Bernstein and setting $t = \frac{1}{4\sqrt{r}}$, we have

$$\mathbb{P}\left[\|\mathcal{P}_{T^\perp}\mathcal{R}_j(F)\| \leq \frac{1}{4\sqrt{r}}\right] \geq 1 - n\exp\left[-\min\left(\frac{q}{96rn_1n_2\kappa(F)}, \frac{q}{8\sqrt{\kappa(F)r}n_1n_2\nu(\mathcal{A})}\right)\right].$$

□

Proof of Lemma 5.8.5

Proof. For a fixed $i \in [m]$, define

$$X_j = \frac{1}{q}\langle F, A_i \rangle - \left\langle A_i, \frac{n_1n_2}{q}\mathcal{P}_T A_j \right\rangle \langle F, A_j \rangle .$$

Then

$$\sum_{j=1}^{q} X_j = \langle (\mathcal{I} - \mathcal{P}_T \mathcal{R} \mathcal{P}_T) F, A_i \rangle .$$

Note that the expectation of X_j with respect to A_j is

$$\mathbb{E}[X_j] = \frac{1}{q} \langle F, A_i \rangle - \frac{1}{q} \langle F, A_i \rangle = 0.$$

And the variance of X_j is bounded above by the variance of the second term

$$\begin{aligned}
\mathbb{E}[X_j^2] &\leq \mathbb{E} \left\langle A_i, \frac{n_1 n_2}{q} \mathcal{P}_T A_j \right\rangle^2 \langle F, A_j \rangle^2 \\
&\leq \frac{n_1^2 n_2^2}{q^2} \kappa(F) \mathbb{E} \langle A_i, \mathcal{P}_T A_j \rangle^2 \\
&= \frac{n_1^2 n_2^2}{q^2} \kappa(F) \frac{1}{n_1 n_2} \|\mathcal{P}_T(A_i)\|_F^2 \\
&\leq \frac{n_1 n_2}{q^2} \kappa(F) \mu(T) =: V_0^2.
\end{aligned}$$

Furthermore,

$$\begin{aligned}
|X_j| &\leq \frac{1}{q} \sqrt{\kappa(F)} + \frac{n_1 n_2}{q} \sqrt{\kappa(F)} \langle A_i, \mathcal{P}_T A_j \rangle \\
&= \frac{1}{q} \sqrt{\kappa(F)} + \frac{n_1 n_2}{q} \sqrt{\kappa(F)} \langle \mathcal{P}_T A_i, \mathcal{P}_T A_j \rangle \\
&= \frac{1}{q} \sqrt{\kappa(F)} + \frac{n_1 n_2}{q} \sqrt{\kappa(F)} \|\mathcal{P}_T A_i\|_F \|\mathcal{P}_T A_j\|_F \\
&\leq \frac{1}{q} \sqrt{\kappa(F)} + \frac{n_1 n_2}{q} \sqrt{\kappa(F)} \mu(T).
\end{aligned}$$

Thus from the Chernoff bound

$$\mathbb{P} \left[|\langle (\mathcal{I} - \mathcal{P}_T \mathcal{R} \mathcal{P}_T) F, A_i \rangle| > \sqrt{t} \right] \leq 2 \exp \left(- \frac{t}{q V_0^2} \right),$$

as long as $\sqrt{t}$ does not exceed

$$\begin{aligned} 2qV_0^2/\left|X_j\right| &= \frac{2n_1n_2}{q}\kappa(F)\mu(T)\frac{q}{\sqrt{\kappa(F)}\left(1+n_1n_2\mu(T)\right)} \\ &= \sqrt{\kappa(F)}\frac{2n_1n_2\mu(T)}{1+n_1n_2\mu(T)} \\ &\geq \sqrt{\kappa(F)}. \end{aligned}$$

The last inequality is due to $\mu(T) \geq \frac{1}{n_1n_2}$ with overwheming probability (see Lemma 5.5.11). Therefore we have

$$\mathbb{P}\left[\left|\langle(\mathcal{I}-\mathcal{P}_T\mathcal{R}\mathcal{P}_T)F, A_i\rangle\right|^2 > \kappa(F)/2\right] \leq 2\exp\left(-\frac{\kappa(F)}{2qV_0^2}\right) = 2\exp\left(\frac{-q}{2n_1n_2\mu(T)}\right).$$

Now taking the standard union bound over $i \in [m]$ we complete the proof.

□

5.8.6 Proof of Lemma 5.5.11

Proof. Our approach is to bound for each i and then take the standard union bound over $i \in [m]$.

Note that

$$\begin{aligned} \|\mathcal{P}_T(A_i)\|_F^2 &\leq \|P_UA_i\|_F^2 + \|A_iP_V\|_F^2 + \|P_UA_iP_V\|_F^2 \\ &\leq 3\max\{\|P_UA_i\|_F^2, \|A_iP_V\|_F^2\}. \end{aligned}$$

Note that $\|P_UA_i\|_F^2$ and $\|A_iP_V\|_F^2$ are almost the same except the dimension. We first bound $\|P_UA_i\|_F^2$

For a particular i, $\|P_U A_i\|_F^2$, we have (neglecting i for simplicity):

$$\begin{aligned}
\|P_U A\|_F^2 &= \left\| \left(\sum_{i=1}^{r} u_i u_i^T \right) A \right\|_F^2 \\
&= \sum_{j=1}^{n_2} \left\| \left(\sum_{i=1}^{r} u_i u_i^T \right) a_j \right\|_F^2 \\
&= \sum_{j=1}^{n_2} \left\| \sum_{i=1}^{r} u_i < u_i, a_j > \right\|_F^2 \\
&= \sum_{j=1}^{n_2} \sum_{i=1}^{r} \| u_i < u_i, a_j > \|_F^2 \quad \text{(Orthogonality of } U\text{)} \\
&= \sum_{j=1}^{n_2} \sum_{i=1}^{r} < u_i, a_j >^2 \quad \text{(Normality of } U\text{)} \\
&= \left\| U^T A \right\|_F^2 ,
\end{aligned}$$

where a_j is the j^{th} column of A, u_i is the i^{th} column of U.

For a simple case, A is normalized random Gaussian matrix, then $\sum_{j=1}^{n_2} \sum_{i=1}^{r} < u_i, a_j >^2$ is $\frac{1}{n_1 n_2} \chi^2(r n_2)$, which is less than $\frac{3r}{n_1}$ with probability exceeding $1 - \exp(-r n_2)$.

We might conjecture that $\|\mathcal{P}_T(A)\|_F^2$ is $\Theta(\frac{r}{n_{\min}})$ with high probability for the RSS ensemble. However such simple argument based on tail bound of Chi-squared distribution does not hold for our model.

Let $S = \left\|U^T A\right\|_F^2$, $\tilde{u}_j$ be the j^{th} row of U, then

$$\begin{aligned}
S &= \sum_{i=1}^{n_2} a_i^T U U^T a_i \\
&= \sum_{i=1}^{n_2} \left(\sum_{j=1}^{n_1} a_{i,j} \tilde{u}_j \right) \left(\sum_{j=1}^{n_1} a_{i,j} \tilde{u}_j^T \right) \\
&= \sum_{i=1}^{n_2} \sum_{h=1}^{r} \left(\sum_{j=1}^{n_1} a_{i,j} \tilde{u}_{j,h} \right) \left(\sum_{j=1}^{n_1} a_{i,j} \tilde{u}_{j,h} \right) \\
&= \sum_{i=1}^{n_2} \sum_{h=1}^{r} \sum_{j=1}^{n_1} \sum_{k=1}^{n_1} a_{i,j} \tilde{u}_{j,h} a_{i,k} \tilde{u}_{k,h} \\
&= \sum_{i=1}^{n_2} \sum_{j=1}^{n_1} \sum_{k=1}^{n_1} a_{i,j} a_{i,k} \sum_{h=1}^{r} \tilde{u}_{j,h} \tilde{u}_{k,h} \\
&= \sum_{i=1}^{n_2} \sum_{j=1}^{n_1} \sum_{k=1}^{n_1} \beta_{i,j,k} \alpha_{j,k},
\end{aligned}$$

where $a_{i,j}$ is th (i,j) entry of matrix A^T, $\tilde{u}_{i,j}$ is the (i,j) entry of U, $\beta_{i,j,k} := a_{i,j} a_{i,k}$ and $\alpha_{j,k} =< \tilde{u}_j, \tilde{u}_k >$.

Note that

$$\begin{aligned}
S &= \sum_{i=1}^{n_2} \sum_{j=1}^{n_1} \beta_{i,j,j} \alpha_{j,j} + \sum_{i=1}^{n_2} \sum_{j=1}^{n_1} \sum_{k=1, k \neq j}^{n_1} \beta_{i,j,k} \alpha_{j,k} \\
&= \underbrace{\sum_{i=1}^{n_2} \sum_{j=1}^{n_1} \sum_{k=1, k \neq j}^{n_1} \beta_{i,j,k} \alpha_{j,k}}_{S_1} + \underbrace{\sum_{i=1}^{n_2} \sum_{j=1}^{n_1} \beta_{i,j,j} \alpha_{j,j}}_{S_2}.
\end{aligned}$$

Since $\mathbb{E}[\beta_{i,j,k}] = 0$ if $j \neq k$ and $\mathbb{E}[\beta_{i,j,k}] = p\delta^2$ if $j = k$, we have that $\mathbb{E}[S_1] = 0$ and $\mathbb{E}[S_2] = r n_2 \delta^2 p = \frac{r}{n_1}$. Now compute $\mathbb{E}[S]$:

$$\mathbb{E}[S] = \mathbb{E}[S_1] + \mathbb{E}[S_2] = \frac{r}{n_1}.$$

Our goal is to show that S concentrates around its expectation $\mathbb{E}[S]$. First we upper bound S_2. Since $S_2 = \sum_{i=1}^{n_2} \sum_{j=1}^{n_1} \beta_{i,j,j} \alpha_{j,j}$ and $\frac{1}{\sigma^2} \beta_{i,j,j} \sim$ Bernoulli(p). Note that $\sum_{j=1}^{n_1} =$

$\alpha_{j,j} = r$.

Using Theorem 5.8.3 we have

$$\mathbb{P}\left\{S_2 \geq (1+\epsilon)\delta^2 rpn_2\right\} \leq \exp\left[-\frac{\epsilon^2 rpn_2}{(2+\epsilon)\max_{j\in[n_1]}\alpha_{j,j}}\right].$$

If $3\max_i \alpha_{i,i}\log n \geq rpn_2$, let let $\epsilon = \frac{9\max_i \alpha_{j,j}\log n}{rpn_2}$, then

$$\mathbb{P}\left\{S_2 \geq 18\delta^2 \max_j \alpha_{j,j}\log n\right\} \leq \frac{3}{n^3}.$$

If $3\max_i \alpha_{i,i}\log n \leq rpn_2$, let let $\epsilon = \sqrt{\frac{9\max_i \alpha_{j,j}\log n}{rpn_2}}$, then

$$\mathbb{P}\left\{S_2 \geq 4rpn_2\delta^2)\right\} \leq \frac{3}{n^3}.$$

In sum

$$\mathbb{P}\left\{S_2 \geq \max\left(4rpn_2\delta^2, 18\delta^2 \max_j \alpha_{j,j}\log n\right)\right\} \leq \frac{3}{n^3}. \tag{5.18}$$

Now turn to S_1. Note that $A = \delta G \circ B$ in (5.10). One can easily rewrite S_1 as

$$S_1 = \delta^2[G_1^T, G_2^T, \ldots G_{n_2}^T]H[G_1^T, G_2^T, \ldots G_{n_2}^T]^T,$$

where G_i is the i^{th} column of G. The matrix H is block diagonal. The r^{th} block is

$$[H]_r = \alpha \circ (B_r B_r^T) - \text{diag}\left(\alpha \circ (B_r B_r^T)\right),$$

where B_i is the i^{th} column of B and $\alpha \in \mathbb{R}^{n_1\times n_1}$ with (i,j) entry of $\alpha_{i,j} =< \tilde{u}_j, \tilde{u}_k >$.

Applying the Hanson-Wright inequality in Theorem 5.8.2 with repsect to the Radamacher variables G, we have

$$\mathbb{P}\left\{|S_1| \geq \delta^2 t\right\} \leq 2\exp\left[-c\min\left(\frac{t^2}{\|H\|_F^2}, \frac{t}{\|H\|_2}\right)\right].$$

Since $\|H\|_2 \leq \|H\|_F$, so

$$\mathbb{P}\left\{|S_1| \geq \delta^2 t\right\} \leq 2\exp\left[-c\min\left(\frac{t^2}{\|H\|_F^2}, \frac{t}{\|H\|_F}\right)\right].$$

Now we bound $\|H\|_F^2$. We have $\|H\|_F^2 = \sum_{r=1}^{n_2} \|[H]_r\|_F^2$. Then

$$\|H\|_F^2 = \sum_{r=1}^{n_2}\sum_{i=1}^{n_1}\sum_{j=1}^{n_1} \alpha_{i,j}^2 B_{i,r}^2 B_{j,r}^2 = \sum_{r=1}^{n_2}\sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1} \alpha_{i,j}^2 B_{i,r}^2 B_{j,r}^2.$$

According to Theorem 5.8.3, we have

$$\mathbb{P}\left\{\|H\|_F^2 \geq (1+\epsilon)p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1} \alpha_{i,j}^2\right\} \leq \exp\left[\frac{-\epsilon^2 p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1} \alpha_{i,j}^2}{(2+\epsilon)\max_{i,j}\alpha_{i,j}^2}\right].$$

If $3\max_{i,j}\alpha_{i,j}^2 \log n \geq p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1} \alpha_{i,j}^2$, let $\epsilon = \frac{9\max_{i,j}\alpha_{i,j}^2 \log n}{p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,j}^2}$, then

$$\mathbb{P}\left\{\|H\|_F^2 \geq 10\max_{i,j}\alpha_{i,j}^2 \log n\right\} \leq \frac{1}{n^3}.$$

If $3\max_{i,j}\alpha_{i,j}^2 \log n < p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1} \alpha_{i,j}^2$, let $\epsilon = \sqrt{\frac{9\max_{i,j}\alpha_{i,j}^2 \log n}{p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,j}^2}}$, then

$$\mathbb{P}\left\{\|H\|_F^2 \geq 4p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1} \alpha_{i,j}^2\right\} \leq \frac{1}{n^3}.$$

In sum

$$\mathbb{P}\left\{\|H\|_F^2 \geq \max\left(4p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1} \alpha_{i,j}^2, 10\max_{i,j}\alpha_{i,j}^2 \log n\right)\right\} \leq \frac{1}{n^3}.$$

To make the probability small enough, i.e.,$\Theta(1/(n^3)$ we set $t = 3\log n\|H\|_F$. Then we

have

$$\mathbb{P}\left\{|S_1| \leq \frac{3\log n}{n_1 n_2 p}\sqrt{\max\left(4p^2 n_2 \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,j}^2, 10\max_{i,j}\alpha_{i,j}^2\log n\right)}\right\} \leq \frac{2}{n^3} \quad (5.19)$$

Combining (5.18) and (5.19) taking the union bound, we have

$$\mathbb{P}\left\{S \leq \frac{C\log n}{n_1 n_2 p}\max\left(p\sqrt{n_2\sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,j}^2}, \max_{i,j}|\alpha_{i,j}|\sqrt{\log n}, \max_i \alpha_{i,i}, \frac{rpn_2}{\log n}\right)\right\} \leq \frac{3}{n^3},$$

where C is some universal positive constant. Since $\max_{i,j}\alpha_{i,j} \leq \max_i \alpha_{i,i}$, so

$$\mathbb{P}\left\{\|P_U A\|_F^2 \leq \frac{C\log n}{n_1 n_2 p}\max\left(p\sqrt{n_2\sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,j}^2}, \max_i \alpha_{i,i}\sqrt{\log n}, \frac{rpn_2}{\log n}\right)\right\} \leq \frac{3}{n^3},$$

Similarly we have

$$\mathbb{P}\left\{\|P_V A\|_F^2 \leq \frac{C\log n}{n_1 n_2 p}\max\left(p\sqrt{n_1\sum_{i=1}^{n_2}\sum_{j=1,j\neq i}^{n_2}\beta_{i,j}^2}, \max_i \beta_{i,i}\sqrt{\log n}, \frac{rpn_1}{\log n}\right)\right\} \leq \frac{3}{n^3},$$

where $\beta_{j,k} =< \tilde{v}_j, \tilde{v}_k >$.

Define $\tau_V := \sum_{i=1}^{n_2}\sum_{j=1,j\neq i}^{n_2}\beta_{i,j}^2$ and $\tau_U := \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,j}^2$. Note that $\tau_U \leq \frac{r^2}{2}$, since τ_U can be bounded as

$$\tau_U \leq \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,i}\alpha_{j,j} \leq \sum_{i=1}^{n_1}\sum_{j=1,j\neq i}^{n_1}\alpha_{i,i}\alpha_{j,j} + \frac{1}{2}\sum_{i=1}^{n_1}\alpha_{i,i}^2 \leq \frac{1}{2}\left(\sum_{i=1}^{n_1}\alpha_{i,i}\right)^2 = \frac{r^2}{2}.$$

Similarly $\tau_V \leq \frac{r^2}{2}$. Then since $n_1 \gtrsim \log n_1$, we have $\frac{rpn_1}{\log n} \gtrsim p\sqrt{n_1\tau_V}$. Similarly $\frac{rpn_2}{\log n} \gtrsim p\sqrt{n_2\tau_U}$.

Now since $\max_i \beta_{i,i} \leq \frac{\gamma_0 r}{n_2}$ and $\max_i \alpha_{i,i} \leq \frac{\gamma_0 r}{n_1}$, taking the union bound we have

$$\mathbb{P}\left\{\|\mathcal{P}_T A\|_F^2 \leq \frac{C\log n}{n_1 n_2 p}\max\left(\frac{\gamma_0 r\sqrt{\log n}}{n_{\min}}, \frac{rpn_{\max}}{\log n}\right)\right\} \leq \frac{6}{n^3},$$

To make the upper bound on $\|\mathcal{P}_T A\|_F^2$ in the order of $\Theta\left(\frac{r\log^2 n}{\sqrt{n_1 n_2}}\right)$, it is sufficient that

$$\gamma_0 \lesssim n_{\min}\sqrt{n_1 n_2}p\sqrt{\log n}.$$

Taking the union bound over $i \in [m]$ and considering $m \lesssim n^2$ we conclude the proof.

□

5.8.7 Proof of Lemma 5.5.12

Proof. Let $F = UV^T/\left\|UV^T\right\|_F$, $f = \text{vec}(F)$, $a_i = \text{vec}(A_i)$, $g_i = \text{vec}(G_i)$ and $b_i = \text{vec}(B_i)$. Then $\langle A_i, F\rangle^2 = a_i^T f f^T a_i$. Compute

$$\mathbb{E}\left[a_i^T f f^T a_i\right] = p\delta^2 \text{tr}(ff^T) = p\delta^2 \|F\|_F^2 = p\delta^2 = \frac{1}{n_1 n_2}.$$

Note that $a_i^T f f^T a_i = \delta^2 g_i^T (f\circ b_i)(f\circ b_i)^T g_i$. Applying Theorem 5.8.2 we have

$$\begin{aligned}&\mathbb{P}\left\{\left|a_i^T f f^T a_i - \frac{1}{n_1 n_2}\right| \geq \delta^2 t\right\}\\ &\leq \exp\left[-c\min\left(\frac{t}{\|(f\circ b_i)(f\circ b_i)^T\|_2}, \frac{t^2}{\|(f\circ b_i)(f\circ b_i)^T\|_F^2}\right)\right]\end{aligned}$$

Also note that $(f\circ b_i)(f\circ b_i)^T$ is rank-1 matrix. So $\left\|(f\circ b_i)(f\circ b_i)^T\right\|_2 = \left\|(f\circ b_i)(f\circ b_i)^T\right\|_F$. And $\left\|(f\circ b_i)(f\circ b_i)^T\right\|_F = \|(f\circ b_i)\|_F^2 = \sum_{j,k} f_{j,k}^2 b_{i,j,k}^2$.

Since $\sum_{j,k} f_{j,k}^2 = 1$, applying theorem 5.8.3 we have

$$\mathbb{P}\left\{\left\|(f\circ b_i)(f\circ b_i)^T\right\|_F \leq (1+\epsilon)p\right\} \leq \exp\left[\frac{-\epsilon^2 p}{(2+\epsilon)\max_{i,j} f_{i,j}^2}\right].$$

If $3\log n \max_{i,j} f_{i,j}^2/p \geq 1$, let $\epsilon = 9\log n \max_{i,j} f_{i,j}^2/p$ and we have

$$\mathbb{P}\left\{\left\|(f\circ b_i)(f\circ b_i)^T\right\|_F \geq 10\log n \max_{i,j} f_{i,j}^2\right\} \leq \frac{1}{n^3}.$$

If $3\log n \max_{i,j} f_{i,j}^2/p \leq 1$, let $\epsilon = \sqrt{9\log n \max_{i,j} f_{i,j}^2/p}$ and we have

$$\mathbb{P}\left\{\left\|(f\circ b_i)(f\circ b_i)^T\right\|_F \geq 4p\right\} \leq \frac{1}{n^3}.$$

In sum

$$\mathbb{P}\left\{\left\|(f\circ b_i)(f\circ b_i)^T\right\|_F \geq \max\left(10\log n \max_{i,j} f_{i,j}^2, 4p\right)\right\} \leq \frac{1}{n^3}.$$

We let $t = \frac{1}{c}\log n \left\|(f\circ b_i)(f\circ b_i)^T\right\|_F$ and take the union bound, then we have

$$\mathbb{P}\left\{\left|a_i^T f f^T a_i - \frac{1}{n_1 n_2}\right| \geq \frac{C\log n}{n_1 n_2 p}\max\left(\log n \max_{i,j} f_{i,j}^2, p\right)\right\} \leq \frac{2}{n^3},$$

where C is some universal positive constant.

Since $\frac{\log n}{n_1 n_2} \gtrsim \frac{1}{n_1 n_2}$ and $\max_{i,j} f_{i,j}^2 = \frac{\gamma_1^2}{n_1 n_2}$, so the above inequality can be simplified as

$$\mathbb{P}\left\{a_i^T f f^T a_i \geq \frac{C\log n}{n_1 n_2 p}\max\left(\frac{\log n}{n_1 n_2}\gamma_1^2, p\right)\right\} \leq \frac{2}{n^3}.$$

To make $a_i^T f f^T a_i$ the order of $\Theta(\frac{\log^2 n}{\sqrt{n_1 n_2}})$, it is sufficient that

$$\gamma_1^2 \lesssim p(n_1 n_2)^{3/2}.$$

Taking the union bound over $i \in [m]$ and considering $m \lesssim n^2$, we complete the proof. □

5.8.8 Proof of Lemma 5.5.13

Proof. First we fix at one particular measurement matrix A (neglecting the subscription i). Before the formal proof, we introduce a theorem on the tail bound of the spectral norm of random matrix.

Theorem 5.8.6 ([[107], Corollary 3.11)**.** *Let X be the $n \times m$ matrix with $X_{i,j} = g_{i,j} b_{i,j}$,*

where $b_{i,j}$ is deterministic parameters and $g_{i,j}$ is centered and sub-Gaussian in the sense

$$\mathbb{E}[g_{i,j}] = 0, \qquad \mathbb{P}\left\{|g_{i,j} > t| \leq Ce^{-t^2/2c}\right\} \quad \textit{for all } t > 0 \textit{ and } i, j.$$

Now define $\sigma_* := \max i, j|b_{i,j}|$, $\sigma_1 := \max_i \sqrt{\sum_j b_{i,j}^2}$ and $\sigma_2 := \max_j \sqrt{\sum_i b_{i,j}^2}$. We have

$$\mathbb{P}\left\{\|X\| \geq (1+\epsilon)\left[\sigma_1 + \sigma_2 + \frac{5}{\sqrt{\log(1+\epsilon)}}\sigma_*\sqrt{\log(n \wedge m)}\right] + t\right\} \leq e^{-t^2/2\delta_*^2}$$

for any $0 < \epsilon \leq 1/2$ and $t \geq 0$. In particular, for every $0 < \epsilon \leq 1/2$ there exists a universal constant c'_ϵ such that for every $t \geq 0$

$$\mathbb{P}\left\{\|X\| \geq (1+\epsilon)(\sigma_1 + \sigma_2) + t\right\} \leq (n \wedge m)e^{-t^2/c'_\epsilon\sigma_*^2}.$$

The basic idea is to apply 5.8.6. First we bound σ_1, σ_1 and σ_*. For simplicity, first ignore the subscription i. It is obvious that $\sigma_* = 1$. Note that that $\sum_j B_{i,j}^2 \sim \text{Binomial}(n2, p)$, so we have

$$\mathbb{P}\left\{\sum_j B_{i,j}^2 \geq (1+\epsilon)n_2 p\right\} \leq e^{\frac{-\epsilon^2 n_2 p}{1+\epsilon}}.$$

So taking the union bound we have

$$\mathbb{P}\left\{\sigma_1^2 \geq (1+\epsilon)n_2 p\right\} \leq n_1 e^{\frac{-\epsilon^2 n_2 p}{1+\epsilon}}.$$

Similarly we have

$$\mathbb{P}\left\{\sigma_2^2 \geq (1+\epsilon)n_1 p\right\} \leq n_1 e^{\frac{-\epsilon^2 n_1 p}{1+\epsilon}}.$$

Now we consider two cases. If $n_2 p > 12 \log n$, let $\epsilon = \sqrt{\frac{12 \log n}{n_2 p}}$ and we have

$$\mathbb{P}\left\{\sigma_1^2 \geq 2n_2 p\right\} \leq \frac{1}{n^3}.$$

If $n_2 p \leq 12 \log n$, let $\epsilon = \frac{12 \log n}{n_2 p}$ and we have

$$\mathbb{P}\left\{\sigma_1^2 \geq 13 \log n\right\} \leq \frac{1}{n^3}$$

In both cases we have

$$\mathbb{P}\left\{\sigma_1^2 \geq \max\left(13 \log n, 2 n_2 p\right)\right\} \leq \frac{1}{n^3}.$$

And similarly

$$\mathbb{P}\left\{\sigma_2^2 \geq \max\left(13 \log n, 2 n_1 p\right)\right\} \leq \frac{1}{n^3}.$$

So taking the Union bound we have

$$\mathbb{P}\left\{\sigma_1 + \sigma_2 \geq \max\left(2\sqrt{13 \log n}, 2\sqrt{2 n_{\max} p}\right)\right\} \leq \frac{2}{n^3}.$$

Let $t = \sqrt{3 c_\epsilon' \log n}$ and substitute to Theorem 5.8.6 and take the union bound, then

$$\mathbb{P}\left\{\|A/\delta\| \geq (1+\epsilon')\left[\max\left(2\sqrt{13 \log n}, 2\sqrt{2 n_{\max} p}\right)\right] + \sqrt{3 c_\epsilon' \log(n_1 + n_2)}\right\} \leq \frac{3}{n^3}$$

Let $\epsilon' = 1/4$, and we can conclude that

$$\mathbb{P}\left\{\|A\| \geq C\delta \max\left(\sqrt{\log n}, \sqrt{n_{\max} p}\right)\right\} \leq \frac{3}{n^3},$$

where C is some universal positive constant.

Taking the union bound over $i \in [m]$ and considering that $m \lesssim n^2$ we complete the proof.

□

CHAPTER 6
CONCLUSIONS AND FUTURE WORK

This thesis primarily studied two prototypical dynamics models in the context of low-rank matrix recovery: random walk dynamics and measurement induced dynamics. We have analyzed these models theoretically and empirically. For the random walk dynamics model, we proposed the locally weighted matrix smoothing (LOWEMS) framework in Chapter 2, establishing error bounds and convergence guarantees for LOWEMS in both the matrix sensing and matrix completion cases. We further provided both synthetic and real-world experimental results to verify our analysis and demonstrate superior empirical performance over static baselines by exploiting dynamic constraints in a recommendation system. In Chapter 3, we discussed two extensions of LOWEMS: one-bit LOWENS for binary measurements and S-LOWEMS for fast simultaneous recovery of a series of low-rank matrices. Our analysis, simulations, and experiments in Chapter 2 and Chapter 3 all demonstrate the effectiveness of weighted matrix smoothing for the problem of low-rank matrix smoothing with random walk dynamics.

However, random walk dynamics is not always the best model for low-rank matrix recovery problems. In many real-world applications, the underlying low-rank matrix is changing according to its interactions with the measurement system, such as in the student learning process. In Chapter 4, we presented the *DynEmb* framework for tracking student knowledge in an ITS from a practical perspective. In Chapter 5, we analyzed theoretically a simple low-rank matrix recovery model which is inspired by the *DynEmb* framework.

Through our theoretical and experimental investigation of dynamic low-rank matrix recovery under these two dynamics models, we demonstrated the effectiveness of exploiting dynamics in low-rank matrix recovery. The LOWEMS framework is effective in various applications (including recommendation systems and personalized learning) compared to

static baselines, and it admits provable recovery guarantees statistically and algorithmically; however, low-rank matrix recovery with a more elaborate dynamics model is still hard to analyze, and designing provably good algorithms for these more complicated models is still challenging. Therefore, we want to highlight several interesting future research directions.

First, a natural extension of the random walk dynamics model is to allow more general linear dynamics. Analysis of the statistical and algorithmic properties of dynamic low-rank matrix recovery with general unknown linear transition dynamics is still not available. Due to the presence of the transition matrix, the applicability of concentration tools from the low-rank matrix recovery literature seems limited. How to handle potential system instability caused by the exponentiation of the transition matrix in this setting is also still unknown. Borrowing techniques from recent finite-sample analysis of stochastic linear dynamical system identification [108, 109, 110] might produce tools that can handle the recurrence of the transition matrix in the context of dynamic low-rank matrix recovery.

Second, it is worthwhile to design more elaborate dynamics models in the *DynEmb* framework, such as LSTMs with an attention mechanism [111]. The attention mechanism mitigates the inability of the LSTM to remember long sequences; this might have benefit in situations where relevant information is far away in time. For example, a student might answer a question correctly with large probability if he did a similar one a month ago, while a one-month-old interaction might be forgotten by an LSTM.

Finally, although analyzing dynamic low-rank matrix recovery with a general dynamic model, (e.g., an LSTM or a GRU) is challenging, there is still some related recent progress, such as a recent generalization analysis on recurrent neural networks [112].

REFERENCES

[1] Z. Liu and L. Vandenberghe, "Interior-point method for nuclear norm approximation with application to system identification," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 3, pp. 1235–1256, 2009.

[2] B. Boots and G. Gordon, "A spectral learning approach to range-only slam," in *International Conference on Machine Learning*, 2013, pp. 19–26.

[3] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE transactions on audio, speech, and language processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[4] M. S. Asif, W. Mantzel, and J. Romberg, "Random channel coding and blind deconvolution," in *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, IEEE, 2009, pp. 1021–1025.

[5] A. Ahmed, B. Recht, and J. Romberg, "Blind deconvolution using convex programming," *IEEE Transactions on Information Theory*, vol. 60, no. 3, pp. 1711–1732, 2014.

[6] G. Tang and B. Recht, "Convex blind deconvolution with random masks," in *Computational Optical Sensing and Imaging*, Optical Society of America, 2014, CW4C–1.

[7] Y. Koren, *The Bellkor solution to the Netflix grand prize*, 2009.

[8] N. Koenigstein, G. Dror, and Y. Koren, "Yahoo! music recommendations: Modeling music ratings with temporal dynamics and item taxonomy," in *Proceedings of the fifth ACM conference on Recommender systems*, ACM, 2011, pp. 165–172.

[9] D. P. Woodruff *et al.*, "Sketching as a tool for numerical linear algebra," *Foundations and Trends® in Theoretical Computer Science*, vol. 10, no. 1–2, pp. 1–157, 2014.

[10] G. Dror, N. Koenigstein, Y. Koren, and M. Weimer, "The Yahoo! music dataset and KDD-Cup'11," in *Proc. ACM SIGKDD Int. Conf. on Knowledge, Discovery, and Data Mining (KDD)*, San Diego, CA, Aug. 2011.

[11] Y. Koren, "Collaborative filtering with temporal dynamics," *Comm. ACM*, vol. 53, no. 4, pp. 89–97, 2010.

[12] N. Mohammadiha, P. Smaragdis, G. Panahandeh, and S. Doclo, "A state-space approach to dynamic nonnegative matrix factorization," *IEEE Trans. Signal Processing*, vol. 63, no. 4, pp. 949–959, 2015.

[13] M. Davenport and J. Romberg, "An overview of low-rank matrix recovery from incomplete observations," *IEEE J. Select. Top. Signal Processing*, vol. 10, no. 4, pp. 608–622, 2016.

[14] Y. Chen and Y. Chi, "Harnessing structures in big data via guaranteed low-rank matrix estimation: Recent theory and fast algorithms via convex and nonconvex optimization," *IEEE Signal Processing Magazine*, vol. 35, no. 4, 2018.

[15] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on information theory*, vol. 52, no. 2, pp. 489–509, 2006.

[16] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE transactions on information theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[17] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on pure and applied mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[18] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE transactions on information theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[19] E. J. Candès *et al.*, "Compressive sampling," in *Proceedings of the international congress of mathematicians*, Madrid, Spain, vol. 3, 2006, pp. 1433–1452.

[20] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[21] ——, "For most large underdetermined systems of linear equations the minimal 1-norm solution is also the sparsest solution," *Communications on pure and applied mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[22] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM journal on computing*, vol. 24, no. 2, pp. 227–234, 1995.

[23] H. L. Taylor, S. C. Banks, and J. F. McCoy, "Deconvolution with the l1 norm," *Geophysics*, vol. 44, no. 1, pp. 39–52, 1979.

[24] E. Candes and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse problems*, vol. 23, no. 3, p. 969, 2007.

[25] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational mathematics*, vol. 12, no. 6, pp. 805–849, 2012.

[26] Y. Gordon, "On milman's inequality and random subspaces which escape through a mesh in r n," in *Geometric Aspects of Functional Analysis*, Springer, 1988, pp. 84–106.

[27] E. J. Candes and Y. Plan, "A probabilistic and ripless theory of compressed sensing," *IEEE Transactions on Information Theory*, vol. 57, no. 11, pp. 7235–7254, 2011.

[28] M.-F. Balcan, Y. Liang, D. P. Woodruff, and H. Zhang, "Matrix completion and related problems via strong duality," in *LIPIcs-Leibniz International Proceedings in Informatics*, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, vol. 94, 2018.

[29] M. Fazel, "Matrix rank minimization with applications," PhD thesis, PhD thesis, Stanford University, 2002.

[30] B. Recht, M. Fazel, and P. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.

[31] E. Candès and Y. Plan, "Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements," *IEEE Trans. Inform. Theory*, vol. 57, no. 4, pp. 2342–2359, 2011.

[32] E. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Inform. Theory*, vol. 56, no. 5, pp. 2053–2080, 2010.

[33] Y. Chen, "Incoherence-optimal matrix completion.," *IEEE Trans. Information Theory*, vol. 61, no. 5, pp. 2909–2923, 2015.

[34] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Transactions on Information Theory*, vol. 56, no. 6, pp. 2980–2998, 2010.

[35] D. Gross, "Recovering low-rank matrices from few coefficients in any basis," *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1548–1566, 2011.

[36] B. Recht, "A simpler approach to matrix completion," *Journal of Machine Learning Research*, vol. 12, no. Dec, pp. 3413–3430, 2011.

[37] E. J. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms," *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1985–2007, 2015.

[38] J. Vinagre, A. M. Jorge, and J. Gama, "An overview on the exploitation of time in collaborative filtering," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 5, no. 5, pp. 195–215, 2015.

[39] P. G. Campos, F. Díez, and I. Cantador, "Time-aware recommender systems: A comprehensive survey and analysis of existing evaluation protocols," *User Modeling and User-Adapted Interaction*, vol. 24, no. 1-2, pp. 67–119, 2014.

[40] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, p. 44, 2014.

[41] K. Hayashi, J.-i. Hirayama, and S. Ishii, "Dynamic exponential family matrix factorization," in *Advances in Knowledge Discovery and Data Mining*, Springer, 2009, pp. 452–462.

[42] Z. Lu, D. Agarwal, and I. S. Dhillon, "A spatio-temporal approach to collaborative filtering," in *Proceedings of the third ACM conference on Recommender systems*, ACM, 2009, pp. 13–20.

[43] Z. Chen and A. Cichocki, "Nonnegative matrix factorization with temporal smoothness and/or spatial decorrelation constraints," *Laboratory for Advanced Brain Signal Processing, RIKEN, Tech. Rep*, vol. 68, 2005.

[44] N. N. Liu, M. Zhao, E. Xiang, and Q. Yang, "Online evolutionary collaborative filtering," in *Proceedings of the fourth ACM conference on Recommender systems*, ACM, 2010, pp. 95–102.

[45] J. Z. Sun, D. Parthasarathy, and K. R. Varshney, "Collaborative kalman filtering for dynamic matrix factorization," *Signal Processing, IEEE Transactions on*, vol. 62, no. 14, pp. 3499–3509, 2014.

[46] L. Xu and M. Davenport, "Dynamic matrix recovery from incomplete observations under an exact low-rank constraint," in *Proc. Adv. in Neural Processing Systems (NIPS)*, Barcelona, Spain, Dec. 2016.

[47] A. Agarwal, S. Negahban, and M. Wainwright, "Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions," *Ann. Stat.*, vol. 40, no. 2, pp. 1171–1197, 2012.

[48] E. Candès and Y. Plan, "Matrix completion with noise," *Proc. IEEE*, vol. 98, no. 6, pp. 925–936, 2010.

[49] E. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[50] M. Davenport, Y. Plan, E. van den Berg, and M. Wootters, "1-bit matrix completion," *Inf. Inference*, vol. 3, no. 3, pp. 189–223, 2014.

[51] O. Klopp, "Noisy low-rank matrix completion with general sampling distribution," *Bernoulli*, vol. 20, no. 1, pp. 282–303, 2014.

[52] S. Negahban and M. Wainwright, "Estimation of (near) low-rank matrices with noise and high-dimensional scaling," *Ann. Stat.*, vol. 39, no. 2, pp. 1069–1097, 2011.

[53] B. Recht, W. Xu, and B. Hassibi, "Necessary and sufficient conditions for success of the nuclear norm heuristic for rank minimization," in *Proc. IEEE Conf. on Decision and Control (CDC)*, Cancun, Mexico, Dec. 2008.

[54] M. Hardt and M. Wootters, "Fast matrix completion without the condition number," in *Proc. Conf. Learning Theory*, Barcelona, Spain, Jun. 2014.

[55] M. Hardt, "Understanding alternating minimization for matrix completion," in *Proc. IEEE Symp. Found. Comp. Science (FOCS)*, Philadelphia, PA, Oct. 2014.

[56] P. Jain and P. Netrapalli, "Fast exact matrix completion with finite samples," in *Proc. Conf. Learning Theory*, Paris, France, Jul. 2015.

[57] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Proc. ACM Symp. Theory of Comput.*, Stanford, CA, Jun. 2013.

[58] R. Keshavan, A. Montanari, and S. Oh, "Matrix completion from noisy entries," in *Proc. Adv. in Neural Processing Systems (NIPS)*, Vancouver, BC, Dec. 2009.

[59] R. Sun and Z.-Q. Luo, "Guaranteed matrix completion via nonconvex factorization," in *Proc. IEEE Symp. Found. Comp. Science (FOCS)*, Berkeley, CA, Oct. 2015.

[60] T. Zhao, Z. Wang, and H. Liu, "A nonconvex optimization framework for low rank matrix estimation," in *Proc. Adv. in Neural Processing Systems (NIPS)*, Montréal, QC, Dec. 2015.

[61] S. Negahban and M. Wainwright, "Restricted strong convexity and weighted matrix completion: Optimal bounds with noise," *J. Machine Learning Research*, vol. 13, no. 1, pp. 1665–1697, 2012.

[62] S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, and B. Recht, "Low-rank solutions of linear matrix equations via procrustes flow," *arXiv preprint arXiv:1507.03566*, 2015.

[63] Q. Zheng and J. Lafferty, "Convergence analysis for rectangular matrix completion using burer-monteiro factorization and gradient descent," *arXiv preprint arXiv:1605.07051*, 2016.

[64] X. Li, J. Lu, R. Arora, J. Haupt, H. Liu, Z. Wang, and T. Zhao, "Symmetry, saddle points, and global optimization landscape of nonconvex matrix factorization," *IEEE Transactions on Information Theory*, vol. 65, no. 6, pp. 3489–3514, 2019.

[65] J. A. Tropp, "An introduction to matrix concentration inequalities," *Found. Trends Mach. Learning*, vol. 8, no. 1–2, pp. 1–230, 2015.

[66] P. Bühlmann and S. Van De Geer, *Statistics for high-dimensional data: Methods, theory and applications*. Springer-Verlag Berlin Heidelberg, 2011.

[67] L. Wang, X. Zhang, and Q. Gu, "A unified computational and statistical framework for nonconvex low-rank matrix estimation," *arXiv preprint arXiv:1610.05275*, 2016.

[68] L. Xu and M. Davenport, "Dynamic one-bit matrix completion," in *Proc. Work. on Signal Processing with Adaptive Sparse Structured Representations (SPARS)*, Lisbon, Portugal, Jun. 2017.

[69] L. Xu and M. A. Davenport, "Simultaneous recovery of a series of low-rank matrices by locally weighted matrix smoothing," in *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, IEEE, 2017, pp. 1–5.

[70] A. Lan, A. Waters, C. Studer, and R. Baraniuk, "Sparse factor analysis for learning and content analytics.," *J. Machine Learning Research*, vol. 15, no. 1, pp. 1959–2008, 2014.

[71] T. Cai and W.-X. Zhou, "A max-norm constrained minimization approach to 1-bit matrix completion," *J. Machine Learning Research*, vol. 14, no. 1, pp. 3619–3647, 2013.

[72] S. Bhaskar and A. Javanmard, "1-bit matrix completion under exact low-rank constraint," in *Proc. IEEE Conf. Inform. Science and Systems (CISS)*, Baltimore, MD, Mar. 2015.

[73] R. Fletcher, "On the Barzilai-Borwein Method," in *Optimization and Control with Applications*, L. Qi, K. Teo, and X. Yang, Eds., Boston, MA: Springer, 2005, pp. 235–256.

[74] Z. Pardos, *Assitment dataset homepage*, `https://sites.google.com/site/assistmentsdata/home/assistment-2009-2010-data`.

[75] A. T. Corbett and J. R. Anderson, "Knowledge tracing: Modeling the acquisition of procedural knowledge," *User modeling and user-adapted interaction*, vol. 4, no. 4, pp. 253–278, 1994.

[76] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein, "Deep knowledge tracing," in *Advances in neural information processing systems*, 2015, pp. 505–513.

[77] H. Cen, K. Koedinger, and B. Junker, "Learning factors analysis–a general method for cognitive model evaluation and improvement," in *International Conference on Intelligent Tutoring Systems*, Springer, 2006, pp. 164–175.

[78] P. I. Pavlik Jr, H. Cen, and K. R. Koedinger, "Performance factors analysis–a new alternative to knowledge tracing.," *Online Submission*, 2009.

[79] J. Zhang, X. Shi, I. King, and D.-Y. Yeung, "Dynamic key-value memory networks for knowledge tracing," in *Proceedings of the 26th international conference on World Wide Web*, International World Wide Web Conferences Steering Committee, 2017, pp. 765–774.

[80] W. van der Linden and R. Hambleton, Eds., *Handbook of Modern Item Reponse Theory*. New York, NY: Springer-Verlag, 2010.

[81] G. Rasch, *Probabilistic Models for Some Intelligence and Attainment Tests*. Copenhagen, Denmark: Danish Institute for Educational Research, 1960.

[82] N. Thai-Nghe, L. Drumond, A. Krohn-Grimberghe, and L. Schmidt-Thieme, "Recommender system for predicting student performance," *Procedia Computer Science*, vol. 1, no. 2, pp. 2811–2819, 2010.

[83] S. Rendle, "Factorization machines," in *2010 IEEE International Conference on Data Mining*, IEEE, 2010, pp. 995–1000.

[84] J.-J. Vie and H. Kashima, "Knowledge tracing machines: Factorization machines for knowledge tracing," *arXiv preprint arXiv:1811.03388*, 2018.

[85] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, vol. 1, no. 2, pp. 270–280, 1989.

[86] X. Xiong, S. Zhao, E. G. Van Inwegen, and J. E. Beck, "Going deeper with deep knowledge tracing.," *International Educational Data Mining Society*, 2016.

[87] K. H. Wilson, Y. Karklin, B. Han, and C. Ekanadham, "Back to the basics: Bayesian extensions of irt outperform neural networks for proficiency estimation," *arXiv preprint arXiv:1604.02336*, 2016.

[88] K. H. Wilson, X. Xiong, M. Khajah, R. V. Lindsey, S. Zhao, Y. Karklin, E. G. Van Inwegen, B. Han, C. Ekanadham, J. E. Beck, *et al.*, "Estimating student proficiency: Deep learning is not the panacea," in *In Neural Information Processing Systems, Workshop on Machine Learning for Education*, 2016, p. 3.

[89] J. González-Brenes, Y. Huang, and P. Brusilovsky, "General features in knowledge tracing to model multiple subskills, temporal item response theory, and expert knowledge," in *The 7th International Conference on Educational Data Mining*, University of Pittsburgh, 2014, pp. 84–91.

[90] M. Khajah, R. Wing, R. Lindsey, and M. Mozer, "Integrating latent-factor and knowledge-tracing models to predict individual differences in learning," in *Educational Data Mining 2014*, Citeseer, 2014.

[91] M. M. Khajah, Y. Huang, J. P. González-Brenes, M. C. Mozer, and P. Brusilovsky, "Integrating knowledge tracing and item response theory: A tale of two frameworks," in *CEUR Workshop Proceedings*, University of Pittsburgh, vol. 1181, 2014, pp. 7–15.

[92] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," *arXiv preprint arXiv:1511.06939*, 2015.

[93] C.-Y. Wu, A. Ahmed, A. Beutel, A. J. Smola, and H. Jing, "Recurrent recommender networks," in *Proceedings of the tenth ACM international conference on web search and data mining*, ACM, 2017, pp. 495–503.

[94] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *2008 Eighth IEEE International Conference on Data Mining*, Ieee, 2008, pp. 263–272.

[95] R. Ge, J. D. Lee, and T. Ma, "Matrix completion has no spurious local minimum," in *Advances in Neural Information Processing Systems*, 2016, pp. 2973–2981.

[96] Y. Qi, D. S. Sachan, M. Felix, S. J. Padmanabhan, and G. Neubig, "When and why are pre-trained word embeddings useful for neural machine translation?" *arXiv preprint arXiv:1804.06323*, 2018.

[97] S. M. Rezaeinia, A. Ghodsi, and R. Rahmani, "Improving the accuracy of pre-trained word embeddings for sentiment analysis," *arXiv preprint arXiv:1711.08609*, 2017.

[98] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, *et al.*, "Wide & deep learning for recommender systems," in *Proceedings of the 1st workshop on deep learning for recommender systems*, ACM, 2016, pp. 7–10.

[99] J. Stamper, A. Niculescu-mizil, S. Ritter, G. G.J Gordon, and K Koedinger, *Challedge data sets from kdd cup 2010*, `https://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp`.

[100] V. Koltchinskii, K. Lounici, A. B. Tsybakov, *et al.*, "Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion," *The Annals of Statistics*, vol. 39, no. 5, pp. 2302–2329, 2011.

[101] Z. Zhu, Q. Li, G. Tang, and M. B. Wakin, "The global optimization geometry of low-rank matrix optimization," *arXiv preprint arXiv:1703.01256*, 2017.

[102] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[103] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *Journal of machine learning research*, vol. 11, no. Aug, pp. 2287–2322, 2010.

[104] W. Yin, "Analysis and generalizations of the linearized bregman method," *SIAM Journal on Imaging Sciences*, vol. 3, no. 4, pp. 856–877, 2010.

[105] M. Rudelson, R. Vershynin, *et al.*, "Hanson-wright inequality and sub-gaussian concentration," *Electronic Communications in Probability*, vol. 18, 2013.

[106] P. Raghavan, "Probabilistic construction of deterministic algorithms: Approximating packing integer programs," *Journal of Computer and System Sciences*, vol. 37, no. 2, pp. 130–143, 1988.

[107] A. S. Bandeira, R. Van Handel, *et al.*, "Sharp nonasymptotic bounds on the norm of random matrices with independent entries," *The Annals of Probability*, vol. 44, no. 4, pp. 2479–2506, 2016.

[108] M. Simchowitz, H. Mania, S. Tu, M. I. Jordan, and B. Recht, "Learning without mixing: Towards a sharp analysis of linear system identification," *arXiv preprint arXiv:1802.08334*, 2018.

[109] T. Sarkar and A. Rakhlin, "How fast can linear dynamical systems be learned?" *arXiv preprint arXiv:1812.01251*, 2018.

[110] A. Tsiamis and G. J. Pappas, "Finite sample analysis of stochastic system identification," *arXiv preprint arXiv:1903.09122*, 2019.

[111] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[112] Z. Allen-Zhu and Y. Li, "Can sgd learn recurrent neural networks with provable generalization?" *arXiv preprint arXiv:1902.01028*, 2019.